

BOX NEW PATENT APPLICATION
Assistant Commissioner for Patents
Washington, D.C. 20231

HAVERSTOCK & OWENS LLP
260 Sheridan Avenue, Suite 420
Palo Alto, California 94306
(650) 833-0160

Attorney Docket No.: ICOM-00700

NEW APPLICATION TRANSMITTAL

Sir:

Transmitted herewith for filing is the patent application of Inventor: Richard S. Neale et al.

Title: GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS

CERTIFICATION UNDER 37 CFR § 1.10

I hereby certify that this New Application and the documents referred to as enclosed herein are being deposited with the United States Postal Service on this date, July 5, 2000, in an envelope bearing "Express Mail Post Office To Addressee" Mailing Label Number EL703214123US addressed to: PATENT APPLICATION, Assistant Commissioner for Patents, Washington, D.C. 20231.

Juan D. Rascon
(Name of Person Mailing Paper)

Signature

Enclosed are:

1. The papers required for filing date under CFR § 1.53(b):

46 Pages of Specification (including claims); 10 Sheet(s) of Drawings.
Formal
Informal

2. X Declaration or Oath (combined with Power of Attorney)

3. X Power of Attorney (combined with Declaration)

4. X Assignment of the Invention to Web Search, Inc (including Form PTO-1595).

Fee Calculation

— Amendment changing number of claims or deleting multiple dependencies is enclosed.

CLAIMS AS FILED

	Number Filed	Number Extra	Rate	Basic Fee
				\$690.00
Total Claims	107 - 20 =	87	\$18.00	1,566.00
Independent Claims	9 - 3 =	6	\$78.00	468.00
Multiple Dependent claim(s), if any			\$260.00	
Filing Fee Calculation				\$2,724.00

X Verified Statement (Declaration) Claiming Small Entity Status (unexecuted)

50% Filing Fee Reduction (if applicable) \$1,362.00

7. Other Fees

X Assignment Recordation Fee 40.00

— Other 0.00

TOTAL FEES ENCLOSED \$1,402.00

8. Payment of Fees

X Check in the amount of \$1,402.00 (\$1,362 for basic filing fee and \$40.00 for Assignment Recordation fee) enclosed.

9. X Authorization to Charge Additional Fees

The Commissioner is hereby authorized to charge any additional fees (or credit any overpayment) associated with this communication and which may be required under 37 CFR § 1.16 or § 1.17 to Account No. 08-1275. An originally executed duplicate of this transmittal is enclosed for this purpose.

10. X Return Receipt Postcard

Dated: July 5, 2000

By: Jonathan O. Owens
Name: Jonathan O. Owens
Registration No.: 37,902

PATENT

Attorney Docket No.: ICOM-00700

Applicant Or Patentee: Richard S. Neale
Serial or Patent No.:
Filed or Issued: herewith
Entitled: GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS

**VERIFIED STATEMENT (DECLARATION) CLAIMING SMALL ENTITY
STATUS (37 CFR § 1.9(c)) - SMALL BUSINESS CONCERN**

I hereby declare that I am

 the owner of the small business concern identified below:
 X an official of the small business concern empowered to act on behalf of the concern identified below:

Name of Concern: **Web Search, Inc.**
Address of Concern: **299 W. Hillcrest Drive, Suite 113
Thousand Oaks, California 91360**

I hereby declare that the above-identified small business concern qualifies as a small business concern as defined in 13 CFR §§ 121.3-18, and reproduced in 37 CFR § 1.9(d), for purposes of paying reduced fees under Sections 41(a) and (b) of Title 35, United States Code, in that the number of employees of the concern, including those of its affiliates, does not exceed 500 persons. For purposes of this statement, (1) the number of employees of the business concern is the average over the previous fiscal year of the concern of the persons employed on a full-time, part-time or temporary basis during each of the pay periods of the fiscal year, and (2) concerns are affiliates of each other when either, directly or indirectly, one concern controls or has the power to control the other, or a third party or parties controls or has the power to control both.

I hereby declare that rights under contract or law have been conveyed to and remain with the small business concern identified above with regard to the invention entitled **GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS** by inventor(s) Richard S. Neale et al. as described in.

 X the specification filed herewith
 Application Serial No. , filed
 Patent No. , issued

If the rights held by the above-identified small business concern are not exclusive, each individual, concern or organization having rights to the invention is listed below and no rights to the invention are held by any person, other than the inventor, who could not qualify as a small business concern under 37 CFR § 1.9(d) or by any concern which would not qualify as a small business concern under 37 CFR § 1.9(d) or a nonprofit organization under 37 CFR § 1.9(e).

*NOTE: Separate verified statements are required from each named person, concern or organization having rights to the invention averring to their status as small entities. (37 CFR § 1.27).

Full Name: _____
Address: _____
[] Individual [] Small Business Concern [] Nonprofit Organization

I acknowledge the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate. (37 CFR § 1.28(b)).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application, any patent issuing thereon, or any patent to which this verified statement is directed.

Name of Person Signing: _____

Title in Organization: _____

Address of Person Signing: Web Search, Inc., 299 W. Hillcrest Drive, Suite 113, Thousand Oaks, California 91360

Dated: _____ Signature: _____

GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS

FIELD OF THE INVENTION:

5 The present invention relates to the field of computers having graphically-oriented user interfaces. More particularly, the present invention relates to the field of constructing Boolean queries and viewing search results through a graphical user interface displayed on a computer.

BACKGROUND OF THE INVENTION:

10 Increased memory and remote electronic data storage capacity offers access to large amounts of data in a very convenient form and physical size. Data may be available on diskette, CD-ROM, magnetic tape, and on line to a centrally located computer and memory storage medium. The challenge remains to extract information from the data simply and efficiently and to have confidence in the result that all relevant items have been uncovered. To focus in on
15 relevant database records, search engines generally use keywords, categorization, segment limitations, Boolean logic, and hit counts. More complex search engines can also employ hierarchical categorization and multifaceted searching.

20 Keywords are the basis of most searches. A simple keyword search, such as that found in most word processors under the "Find" command, will locate the occurrence of a text string within a document or a record. Misspellings, synonyms, or different tenses of a given text string will not be located. The searcher must be cautious to truncate the text string to a word's root. A search for the text string "graphical", for instance, will not locate instances of the text string "graphics." The searcher must also not choose commonly occurring words, as such a search would result in a high number of search results. Keywords are commonly combined with
25 categorization, segment limitations, Boolean logic operators, advanced keywording, date operators and numeric operators to create a more effective search.

30 Categorization is a technique used to focus the scope of a search. A category is a subset of records. By conducting a search only within this subset of records, fewer irrelevant hits result. Lexis-Nexis™ and Dialog™, two online searchable databases with proprietary search engines, are examples of categorized databases. Prior to conducting a keyword search within the Lexis-Nexis™ or Dialog™ database, the searcher must select from an extensive list of categories. Some categories are broader than others. If the searcher selects an overly broad category, his or

her search will result in too many irrelevant hits and the searcher will waste time sorting through the undesired search result records looking for relevant hits. If the searcher selects an overly narrow category, his or her search results will not include some of the desired records. Selection of an appropriate category, therefore, is of vital importance.

5 Searches can be further focused with the use of segment limitations. "Segments" are similar to categories in that they are domain specific. Category classifications are used to divide multiple records into subsets, or "fields". Segment classifications are used to divide individual records into specific groupings of information. Using segments, keyword searches can be targeted at certain fields of a record, such as a record's title or author. Search engines distributed
10 by Lexis-NexisTM and DialogTM, two online searchable database providers, are well-adapted to such targeted searches, often using dozens of segments for each category of records. A news article record, for instance, is typically broken down into separate fields for byline, date, publisher, abstract, and body. To find a news article with the word "elephant" in the title (or headline) using the classical interface of the Lexis-NexisTM search engine, the following syntax would be needed: "HEADLINE(elephant)".
15

Using keyword searching may not be very helpful if the user is not familiar with the appropriate standard terminology related to the information they are looking for. Further, there may be many appropriate ways to describe the information sought by the user. A concept expressed by a standard industry term in one industry may be different from a standard industry term in a different industry. A keyword search would require searching all synonyms used in
20 order to ensure a complete and accurate result.

When a user of a searching/retrieval system enters a keyword search query into a system, the query is parsed. Based on the parsed query, a listing of documents relevant to the query is provided to the user. In the prior art, it is also known to use semantic networks when parsing a
25 query. The number of words used to search the database is then expanded by including the corresponding words or associated words identified by the semantic network in the search instructions. This expansion can be based on any one or a combination of using stems or roots of terms, using sound-a-like words, using wildcard words or any other appropriate semantic technique.

30 Boolean operators, such as "AND", "OR" and "MINUS", are used to enhance the capabilities of a search engine. The basic format of Boolean queries is well known in the art and generally takes on the form of "X OR Y", where X and Y are two distinct keywords. Because

In the above example, the hit counts are as follows: in the database the term “cat” is included in 280 records; in the database the term “dog” is included in 494 records; in the database the term “veterinarian” is included in 34 records; in the database the term “doctor” is included in 194 records; in the database the term “cat” or “dog” is included in 774 records; in the database the term “veterinarian” or “doctor” is included in 228 records; and in the database the Boolean query for the Boolean expression “(cat OR dog) AND (doctor OR veterinarian)” results in the location of 4 records. If the Boolean expression is altered by the replacement of “dog” with “cow”, the hit count change ripples through the Boolean expression’s representation as shown in Boolean Representation Two, illustrated below in Table II.

```

cat -- 280 -----
cow -- 71 -----
veterinarian -- 34 ----
doctor -- 194 -----
                                |-- OR -- 351
                                |
                                |-- AND -- 1
                                |
                                |-- OR -- 228

```

Table II: Boolean Representation Two

Feedback from individual hit counts gives the searcher access to information normally hidden. Viewing individual hit counts, a searcher is better able to identify search terms that are too specific, too broad, or misspelled.

An additional search tool is hierarchical categorization. Instead of classifying records into separate categories, hierarchical categories classify records into both broad groupings and progressively narrower groupings. An example of hierarchical categorization is found in biology, where organisms are organized, from broadest to narrowest, by kingdom, phylum, class, order, family, genus, and species. Hierarchical categorization is commonly used in conventional internet search engines, such as those found at the Yahoo!™ and Altavista™ websites. To find information about a specific topic, a search engine user navigates from a list of broad categories through an increasingly more specific list of categories. Once the first category is selected, a search engine typically displays a lower level screen with another list of alternatives. Such navigation continues down through the various menus of alternatives having decreasing priority levels. At any point of the category navigation, a keyword or Boolean search can be performed upon the records in that category. Search results are only obtained from records located within

the category searched. Most search engines only allow searches in one category at a time. To search a second category, the searcher must navigate up the hierarchical category tree and then down to the second category.

The limits of the hierarchical categorization method were addressed in the early 1990's by the Software Technology for Adaptable, Reliable Systems (STARS™) program, which was spearheaded by International Business Machines Corporation and the Boeing Company. One objective of the STARS™ program was to improve the classification system for software so that previously developed software could be reused in new software development efforts. One proposal resulting from the STARS™ program was multifaceted classification. Instead of assigning a record to a single category, multifaceted classification allows a record to belong to multiple categories. The multiple categories become part of a record's description, along with standard information for the record such as the title, the abstract (or keywords), the date, and author. Multifaceted classification improves the likelihood of locating relevant records. First, the searcher can take several different paths to locate the same record. Using the analogy of books in a library, multifaceted classification is able to place a single book on more than one shelf. Second, the multiple categories can be subjected to a Boolean query. Records relating to sports medicine could be found by searching for records included in both the sports category and the medicine category.

An internet search engine employing multifaceted classification has been developed by the NCBI (National Center for Biological Information), a division of the NLM (National Library of Medicine) at the NIH (National Institutes of Health) for the PubMed database of bibliographic information. The NCBI search engine includes a hierarchical category tree from which categories can be selected. The NCBI search engine permits the searcher to select multiple categories, entitled "MeSH Terms", from a hierarchical category tree. A MeSH Term can be linked by a logical AND or a logical OR with other MeSH Terms to create a Boolean expression. The Boolean expression of MeSH Terms can then be combined with additional terms to create the final query. The NCBI search engine also displays the hit count for each category and for each Boolean combination of categories.

Boolean logic, segment limitations, hit counts, hierarchical categorization, and multifaceted classification help the searcher create more effective queries, but at the cost of increased complexity. Often instruction manuals or a software program's help menu must be consulted to draft a query. Dialog™, for instance, publishes a "Bluebook" that contains detailed

lists of segment codes for each of their many databases. Lexis-Nexis™ goes so far as to provide free online access and training seminars for students to overcome their search engine's initial learning curve. Addressing the complexity of search syntax, efforts have been made in the design of search engine software to reduce the amount of knowledge and experience needed to draft queries. The Lexis-Nexis™ search engine, for instance, provides searchers with the option of using a graphical user interface rather than their classical interface.

Most modern computer systems employ a graphical user interface rather than the more basic textual interface. In a graphical user interface, the user can run application programs, manipulate files, and perform most other necessary functions by manipulating images on the computer's display. This manipulation is accomplished by using cursor control keys and other keyboard keys or by using a cursor controlling peripheral device such as a joystick, mouse or trackball. A computer system 400 with a graphical user interface can be implemented as illustrated in FIG. 1. In FIG. 1, the computer system 400 includes a central processor unit (CPU) 401, a main memory 402, a video memory 403, a keyboard 404 for user input, supplemented by a conventional mouse 405 for manipulating graphic images as a cursor control device and a mass storage device 406, all coupled together by a conventional bidirectional system bus 407. The mass storage device 406 may include both fixed and removable media using any one or more of magnetic, optical or magneto-optical storage technology or any other available mass storage technology. The system bus 407 contains an address bus for addressing any portion of the memory 402 and 403. The system bus 407 also includes a data bus for transferring data between and among the CPU 401, the main memory 402, the video memory 403 and the mass storage device 406. Coupled to a port of the video memory 403 is a video multiplex and shifter circuit 408, to which in turn a video amplifier 409 is coupled. The video amplifier 409 drives a monitor or display 410 on which a graphical user interface is displayed. The video multiplex and shifter circuitry 408 and the video amplifier 409 convert pixel data stored in the video memory 403 to raster signals suitable for use by the monitor 410.

Graphical user interfaces for search engines use one or more screens to assist the searcher in the creation of a query. A sample query input screen 100 is illustrated in FIG. 2. The input screen 100 includes several labeled boxes capable of receiving textual inputs, including a client text box 102, a category text box 104, and a query text box 108. The client text box 102 is included such that individual searches can be billed to different clients and/or different projects. The category text box 104 is provided such that the searcher can input the category that will be

searched. The query text box 108 is provided for the text of the query that will be executed in the selected category. The date parameters for a search are inputted using three boxes: a date parameter box 116, a start date box 120, and an end date box 122. Queries can be executed using the search button 114, saved using the save query button 124, or closed using the close button 126. The searcher can obtain assistance in use of the input screen 100 by pressing the help button 128.

Buttons in the FIG. 2 input screen 100 prompt the display of additional information. A date parameter selection button 118 is used to display available date parameters, such as: "Date Is", "Date After", "Date Before", "Date Between", or "No Date Restriction". To find a record published between 1985 and 1989, the searcher would select "Date Between" for the date parameter box 116, "1985" for the start date box 120, and "1989" for the end date box 122. A segments button 110 displays the segment limitation available for the category selected in the category text box 104. A Boolean operators button 112 displays the Boolean operators available for connecting keywords in the query text box 108, such as: "OR", "AND", "NOT", and the special syntax used for proximity searches. A category walker button 106 is used to choose from a wide selection of available categories.

An exemplary category walker 140 is illustrated in FIG. 3. The category walker 140 includes a category tree viewing area 152, a category scroll bar 154, an OK button 146, a cancel button 148, and a category help button 150. The category viewing area 152 includes a sample hierarchical category tree represented graphically in a folder structure 142 with a root directory 144 and subdirectories arranged in alphabetical order. To navigate through the folder structure 142, the searcher scrolls through the alphabetical list of categories and clicks on the folder icons to view specific subdirectories of the selected category. To select a category, the searcher clicks the folder icon associated with the desired category using the conventional mouse 405 and then clicks the OK button 146. Once the OK button 146 is clicked, the category walker 140 closes and the category tree for the selected category appears in the category text box 104.

Saving a query is different from saving search results. A saved query contains all the information displayed on the input screen 100. When the searcher reopens a saved query, the query is displayed in the input screen 100 in the same manner it was originally displayed. The client information is displayed in the client text box 102, the category is displayed in the category text box 104, the query is displayed in the query text box 108, and any date information is displayed in the date parameter box 116, the start date box 120, and the end date box 122. The

saved query displayed in the input screen 100 can be modified by the searcher and saved under another file name. Search result records are typically not saved in a format that can be manipulated by the search engine. Instead, search engines permit lists of search result records to be downloaded and saved as text documents. Individual search results can also be downloaded and saved as text documents. Once downloaded and saved on the searcher's computer, the list of search results can be manipulated using word processing software. To browse through the search results of a previously executed query using a search engine, a searcher must reexecute a saved query.

The database for a search engine can be local or remote. Local databases are generally marketed in the form of a CD-ROM accompanied by a proprietary search engine tailored for use with the particular data. Once a query is drafted, the proprietary software accesses records from the CD-ROM and displays the relevant records. The CD-ROMs can be used within a stand-alone computer or on a local area network accessible to multiple computers. As a CD-ROM is stored locally, use of the database does not require access to external transmission networks (e.g., telephone lines, ISDN, T-1, or DSL). Avoiding the need for external data, data from local databases can be retrieved faster and with greater reliability. CD-ROMs are not practical in some applications, however. CD-ROMs hold a limited amount of data, so they are not practical for databases that will not fit on a single CD-ROM (about 650 megabytes). The availability of CD-ROMs is also limited. Only widely used databases are available through normal marketing channels. Purchase of CD-ROMs is also impractical when the same information is available to the public over the internet.

Remote databases are searched in much the same way as a local database, but over a communication line. Remote access allows large databases to be centrally located and maintained, resulting in larger storage capability and lower costs. The remote database's need for transmission of data over a communication line, however, results in slower and less reliable retrieval of database information. Before the widespread growth of the internet, most remote databases were accessed by dial-up modems through a direct connection over a telephone line using proprietary data transfer protocols. With the rapid growth of the internet, more database services are accessible through the internet with a typical internet browser, such as Microsoft Internet Explorer or Netscape Communicator.

Access to remote databases through the internet follows the client-server model. The "client" is the searcher's computer. The server is the hardware and software maintained in a

different location by the database provider. FIG. 4 illustrates how the client-server model is typically implemented. FIG. 4 includes a client 170, an internet 172, middleware 176, a first database 184, a second database 186, a third database 188, and a third party web server 199. The client 170 includes a browser 196 and a modem 198. The client 170 is coupled to the
5 middleware 176 through a first communications route 174 traveling through the internet 172. The client 170 is also coupled to the third party web server 199 through a second communications route 197 through the internet 172. The middleware 176 is coupled to the first database 184 through a first database connection 190, to the second database 186 through a second database connection 192, and to a third database 188 through a third database connection
10 194.

The middleware 176 is used to connect the client 170 to database records. The middleware 176 includes a web server 178, a servlet 180, and a Java database connectivity (JDBC) layer 182. The web server 178 is used to control data transfer sent to and arriving from the client 170. Data is transferred between the client 170 and the middleware 176 by one of
15 several data transfer protocols: hypertext markup language (HTML), simple text markup language (STML), extended HTML (XML), remote method invocation (RMI), common object request broker application (CORBA), or a proprietary protocol. The web server 178 receives data arriving from the client 170 and formats the data for the servlet 180. The web server 178 also receives data from the servlet 180, formats the data for the client 170, and sends the data to
20 the client 170 through the internet 172. The servlet 180 is a Java program that receives the requests from the client 170, collects and processes the information requested, and then sends the information to the client 170 through the web server 178. The servlet 180 accesses databases through a JDBC layer 182. The JDBC layer 182 is a programming interface used to access information contained in the databases.

A search using the client-server model begins with a visit to the database provider's home page and ends with a visit to the third party web server 199. The database provider's home page includes instructions on where and how to conduct a search. The Yahoo!™ home page, for instance, includes a hierarchical category index, a box for entering a query, and a search button. Each of the categories listed on a given page is a hyperlink that will request a web page with
25 subcategories for the selected category. Once satisfied with the category choice, the searcher types in a query and clicks the search button. The search request is formatted by the browser 196 and transmitted by the modem 198 across the internet 172 to the middleware 176. The web
30

server 178 formats the request for the servlet 180, which sends a query to a given database through the JDBC layer 182. Records within the Yahoo!™ databases include a category field, a title field, an abstract field, and a universal resource locator (URL) field. The query sent to the database includes the category and keywords for the search. The database searches through the category field and keyword field for matching records. Search results are delivered back to the servlet 180 through the JDBC layer 182. The search results include a hit count total and summaries of each hit including the title, a short description, and a URL. After collecting the search results from a given database, the servlet 180 creates a new web page that is sent to the client through the web server 178 and over the internet 172. The client 170 receives the new web page from the internet 172 through the modem 198 and displays the new web page on the browser 196. The new web page displays the title, short description, and URL for each hit. The title of each hit is a hyperlink to the URL of the corresponding web page. By clicking the hyperlink, the browser is instructed to retrieve the web page found at the selected URL. The web page is then retrieved through the internet 172 from the third party web server 199 and displayed by the browser 196.

The design of internet search pages is constrained by the need to continually retrieve new web pages. Web pages are transmitted to the client 170 as HTML, which includes all the text and graphics via links displayed by the browser 196. The text and graphics displayed by the browser 196 cannot usually be altered unless a new page is requested and received over the internet. Each click of the computer mouse results in a noticeable delay, the length of which depends on the transmission speed of the modem 198 and the efficiency of the middleware 176. This time delay has resulted in the design of internet search engines that require fewer web pages for the execution of a search. The time delay has also resulted in fewer features, as each new feature requires the download of additional web pages for execution. While simplicity has resulted in fewer time delays, it has also resulted in less compelling graphical user interfaces. Each search looks the same as any other search.

SUMMARY OF THE INVENTION:

A graphical user interface for building Boolean queries and viewing search results of the present invention preferably includes a workspace, a toolbar, and a menu bar. Within the workspace, a Boolean query is drafted using search cells and combination cells. The search cells each preferably have a search by icon, a text field, a hit count, a search cell flag, and a negate

icon. The search cells are used to search records by category, keyword, numeric or date expressions. Categories are chosen with the assistance of a category walker. The combination cells each preferably have a combine icon, an expression field, and a hit count. Search cells and combination cells are moved about the workspace by a drag and drop operation of the computer mouse. The combination cell is a Boolean combination of two or more selected search cells and its expression field provides a textual representation of a Boolean combination. Combination cells combine the queries contained in individual search cells with logical AND operations, logical OR operations or logical MINUS operations.

The toolbar includes a search by group, a combine group, and an actions group. The search by group has a category button, a keyword button, and a date button. The category button, when selected, generates one search cell that searches by category with the assistance of the category walker. The keyword button, when selected, generates one search cell that searches by keyword. The date button, when selected, generates one search cell that searches by date. The combine group has a logical AND button, a logical OR button and a logical MINUS button. The logical AND button, when selected, generates one combination cell that combines selected search cells with a logical AND operation. The logical OR button, when selected, generates one combination cell that combines selected search cells with a logical OR operation. The logical MINUS button, when selected, generates one combination cell that combines selected search cells with a logical MINUS operation. The actions group has a remove button, a results button, an expand/collapse button and a quit button. The remove button, when selected, removes selected combination cells and selected search cells from the workspace. The results button, when selected, executes the query defined by the selected combination cell and creates a results tab. The expand/collapse button collapses search cells together into a combination search cell or expands a selected combination search cell into the original search cells which were utilized to generate the combination cell. The quit button, when selected, closes the window. When the results tab is selected, the workspace displays the search results, including a title, a media type, and a location. In the preferred embodiment of the invention, the location of records is preferably a URL address.

The menu bar includes a session menu, a collection menu, a query menu, a select menu, and a help menu. A session treats the query represented in the workspace as a computer file. The session menu includes a new session option, an open session option, a save session option, a close session option, and an exit option. A collection treats the search results represented in the

workspace as a computer file. The collection menu includes a new collection option, an open collection option, a merge collection option, a save option, and a close collection option. The query menu includes a search by category option, a search by keyword option, a search by numeric or date option, a combine by logical AND option, a combine by logical OR option, a combine by logical MINUS option and an execute search option. The select menu includes a select all option, an unselect all option, and a selection option. The help menu includes an index option and an about option.

In the preferred embodiment of the invention, the Boolean logic is depicted with the use of combination brackets. The workspace of the preferred embodiment includes two columns. The left column includes a stack of search cells; the right column is an area for the combination brackets or braces. The combination brackets graphically represent Boolean combinations of search cells. Each combination bracket is labeled with a logical AND, a logical OR or a logical MINUS. The contents of each combination bracket is controllable by the use of movable graphical pins that are located between each pair of search cells. The combination bracket can be represented by use of a rectangular bracket, a square bracket, or a curly brace.

In a further alternative embodiment of the invention, the Boolean query is depicted in a folder tree hierarchy. Search folders and combination folders are used to depict search cells and combination cells, respectively. Search folders contain the same information as search cells, but search folders can be toggled to appear as small folder icons within the workspace. Combination folders are represented as small folder icons containing a logical AND icon, a logical OR icon or a logical MINUS icon. Search folders are shown to be included within combination folders through the use of the folder tree hierarchy, just as computer files are shown to be included within a folder in a typical file manager program, such as Microsoft Windows Explorer™. Search folders and combination folders can be moved about the workspace using a drag and drop motion of the computer mouse.

In the preferred embodiment of the invention, a local client computer uses an internet browser and is coupled to a remote server through the internet. To minimize communication delays across the internet, an applet is run on the local client to perform most graphical display manipulations prompted by the searcher's manipulation of the search cells, the combination cells, the toolbar, and the menu. The applet, for instance, processes and displays the graphics depicting insertion, flagging, movement, and deletion of search cells within the workspace. The remote

server is then used to execute queries received over the internet from the local client and deliver search results back to the local client over the internet.

BRIEF DESCRIPTION OF THE DRAWINGS:

5 FIG. 1 illustrates a computer system with a graphical user interface, including a CRT, a keyboard, and a computer mouse.

 FIG. 2 illustrates a graphical user interface window used to assist a searcher in the creation of a query.

10 FIG. 3 illustrates a category walker designed to assist in the navigation through a category tree.

 FIG. 4 illustrates an implementation of the client-server model whereby a client is linked to databases through the internet and middleware.

15 FIG. 5 illustrates a first window for graphical representation of a Boolean query using search cells and combination cells of the present invention.

 FIG. 6 illustrates a first window for display of search results within a plurality of result cells of the present invention.

 FIG. 7 illustrates a second window for graphical representation of a Boolean query using search cells and a Boolean graphic of the present invention.

20 FIG. 8A to 8H detail how the Boolean graphic of the second window is manipulated to generate a hierarchical Boolean query of the present invention.

 FIG. 9 illustrates a third window that includes a Boolean folder tree used to generate the hierarchical Boolean query of the present invention.

 FIG. 10 illustrates a search cell with a selected advanced button in which the additional features are shown.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT:

25 The graphical user interface for building Boolean queries and viewing search results of the present invention is an improved implementation of a search engine wherein the search terms and Boolean logic making up a Boolean query are displayed in a graphical format. The preferred embodiment includes two window designs for graphically representing Boolean queries. The graphical user interface of the present invention is capable for use on both a stand alone
30 computer and in a client-server model. When used in a client-server model, the search engine of

the present invention employs an applet operating on the client computer system to locally modify the client's graphical display in response to computer mouse and keyboard commands.

The first window 300 for graphically representing a Boolean query in the graphical user interface of the present invention is illustrated in FIG. 5. The first window 300 preferably includes three main areas: a menu bar 301 along the top of the first window 300, a first tool bar 303 along the left side of the first window 300, and a first workspace 336. The menu bar 301 preferably includes a session menu 302, a collection menu 304, a query menu 306, a select menu 308, and a help menu 310. The first tool bar 303 preferably includes three tool groups: a search by group 314, a combine group 322, and an actions group 328. The first tool bar 303 also preferably includes a company logo 312. The search by group 314 includes a category button 316, a keyword button 318, and a date button 320. The combine group 322 includes a logical AND button 324, a logical OR button 326 and a logical MINUS button 327. The actions group 328 includes a remove button 330, a results button 332, an expand/collapse button 333 and a quit button 334. The first workspace 336 includes a query tab 338, a search cell A 340, a search cell B 342, and a combination cell 344. The search cell A 340 and the search cell B 342 both include a text field 348, a negate icon 350, a hit count 352, a search by icon 346, and a cell flag 354. The combination cell 344 includes an expression field 358 and a combine icon 366. Like the search cells 340 and 342, the combination cell also includes: a negate icon 350, a hit count 352 and a cell flag 354.

The negate icon 350 is a check box, which toggles between an activated state and a deactivated state by a click with the computer mouse. The cell flag 354 of a specific cell is activated by clicking on the cell perimeter with the computer mouse and deactivated by a second click. The cell flag 354 indicates whether or not a corresponding search or combination cell is selected. The search by icon 346 within the search cells 340 and 342 indicates the field that is currently being searched. Separate icons are used for category, keyword, and date. Specific categories are chosen by first clicking the category button 316 and then choosing a specific category from a hierarchical category tree using a category walker. Media type is chosen using the category walker. The combine icon 366 indicates whether the combination cell 344 is combining search cells using a logical OR operator, a logical AND operator or a logical MINUS operator. The preferred pull-down options of each of the menus within the menu bar are set forth below in Table III.

Menu Item	Pull-Down Options
Session Menu	New
	Open
	Save
	Close
	Exit
Collection Menu	New
	Open
	Merge
	Save
	Close
Query Menu	by Category
	by Keyword
	by Date
	CombineA.B.
	Execute
Select	Select All
	Unselect All
	Invert Selection
Help Menu	Index
	About

Table III: Pull-Down Options of Menu Bar

A query is created in the first window 300, starting with a blank first workspace 336. The blank first workspace 336 is created using the “New” pull-down option from the session menu 302 (and also when the first search window 300 is first opened). Search cells are added to the first workspace 336 one at a time using the category button 316, the keyword button 318, or the date button 320. Selecting the category button 316 opens a search cell with a category icon as

the search by icon 346, selecting the keyword button 318 opens a search cell with a keyword icon as the search by icon 346, and selecting the date button 320 opens a search cell with a date icon as the search by icon 346. When first opened a new search cell has a blank text field 348, unchecked negate icon 350, and a zero hit count 352. If the search cell is created using the category button 316, a category walker opens in a small, separate window. The category is chosen from navigating through the category tree using the category walker, selecting a given category, and pressing the OK button. The chosen category is automatically placed within the text field 348. The searcher checks the negate icon 350 if he or she desires to locate every record outside the chosen category. Otherwise, if the negate icon 350 is not checked then only records within the category will be included in the search.

If the search cell is created using the keyword button 318, the searcher must enter single keywords into the text field 348. The searcher checks the negate icon 350 if he or she desires to locate every record not containing the given keyword. Otherwise, if the negate icon 350 is not checked then only records containing the keyword will be located. If the search cell is created using the date button 320, a search cell containing a drop down box and two date fields is created. The drop down box allows the searcher to select among the following choices: no date restriction, date is, date after, date before, date between. No dates need be entered in either of the two date fields if "no date restriction" is chosen. Only the first date field is used if the selected drop down option is either "date is," "date after," or "date before." If the "date between" option is chosen, the first date field is used as the starting date and the second date field is used as the ending date. Each search cell has a separate hit count 352. The searcher checks the negate icon 350 if he or she desires to locate every record outside the given date range. Otherwise, if the negate icon 350 is not checked then only records within the date range will be located.

Search cells can be combined into a Boolean query using the combination cell 344. Selecting the logical AND button 324 creates a combination cell with a logical AND combine icon 366. Selecting the logical OR button 326 creates a combination cell with a logical OR combine icon 366. Selecting the logical MINUS button 327 creates a combination cell with a logical MINUS combine icon 366. The expression field 358 of a combination cell 344 is dictated by the content of the search cells that are flagged prior to selecting the logical AND button 324, the logical OR button 326 or the logical MINUS button 327. If "A" and "B" represent selected search cells, the content of the expression field 358 for a combination cell created with a logical AND button 324 is of the form: "A" AND "B." If "A" and "B" represent

selected search cells, the content of the expression field 358 for a combination cell created with a logical OR button 326 is of the form: "A" OR "B." If "A" and "B" represent selected search cells, the content of the expression field 358 for a combination cell created with a logical MINUS button 327 is of the form "A" MINUS "B." The selected search cells are removed from the first workspace 336 upon the creation of the combination cell 344 when the collapse option button 333 is selected, giving the appearance that the search cells have collapsed into the combination cell 344. After a combination cell 344 has been collapsed, it can be expanded by selecting the combination cell 344 and then selecting the expand option button 333 which will expand the combination cell into the original search cells which were used to create the combination cell. Once a combination cell 344 is created, it can be combined with other search cells (or other combination cells) to create an additional combination cell.

The first workspace 336 treats queries as a file. Like a word processing file, queries within the first workspace 336 can be saved, closed, opened, and inserted within other queries. Queries are saved by selecting "Save" from the session menu 302. Queries can be closed by selecting "Close" from the session menu 302 and reopened by selecting "Open." Selecting "Insert" from the session menu 302 opens a dialog box from which the searcher can import a saved search into the present query. This method allows the searcher to combine the query represented on the first workspace 336 with a previously drafted query. Once the previously drafted query is inserted into the first workspace 336, the searcher can delete unwanted search cells or combination cells to create a third query. Cells, and any associated search results, can be deleted from the first workspace 336 by first flagging the cells then selecting the remove button 330. Cells can be moved within the first workspace 336 by selecting a cell with the computer mouse and dragging the cell to a new location. Movement of cells, while not affecting the outcome of a query, is helpful in the visualization of the query's content. Moving related cells such that they are adjacent prior to creating a combination cell, for instance, assists the searcher in the selection of the cells. The searcher may also desire to place two cells next to one another in order to compare and contrast search terms in light of hit counts.

To execute a query in the first workspace 336, the searcher first selects an appropriate cell then selects the results button 332. When the search is executed, the results from that search will be displayed in the first workspace 336. Executing a search causes a change in the appearance of the workspace 336. As shown in FIG. 6, executing a search changes the appearance of the first workspace 336, but preferably does not change the appearance of the menu bar 301 or the first

tool bar 303. In the first workspace 336, each executed search has a results tab 380 and a plurality of result cells. The number of result cells is equal to the hit count 352 of the search cell or combination cell flagged for search execution. In the example illustrated in FIG. 6, result cell one 382, result cell two 384, result cell three 386, and result cell four 388 are displayed.

Additional result cells can be viewed in the first workspace 336, if available, by scrolling down the window of the first workspace 336 using the results scroll bar 398. Each result cell includes a media type icon 390, a title 392, a description 394, a location 396, and a cell flag 354. The media type icon 390 is used to designate the type of record, such as text, jpg, gif, tiff, or pdf. The location 396 designates the URL of the record if the record is located remotely on the world wide web or the file name if the record is located locally on a CD-ROM.

The first workspace 336 treats search results as a file, which is referred to as a "collection." Collections can be saved, closed, opened, and merged within other collections. Collections are saved by selecting "Save" from the collection menu 304. Collections can be closed by selecting "Close" from the collection menu 304 and reopened by selecting "Open." Selecting "Merge" from the collection menu 304 opens a dialog box from which the searcher can import search results from another collection into the present collection. This method allows the searcher to combine the collection represented within the first workspace 336 with a previously executed search collection. Once the previously drafted collection is merged into the first workspace 336, the searcher can delete unwanted result cells to create a third collection. Result cells can be deleted from the first workspace 336 by first selecting the desired result cells then selecting the remove button 330. Result cells can be moved and reordered within the first workspace 336 by selecting a result cell with the computer mouse and dragging the result cell to a new location. Movement of result cells, while not affecting the content of a collection, is helpful in the visualization of the collection's scope. Moving related result cells such that they are adjacent prior to download, for instance, assists the searcher in the organization of the downloaded records. The searcher may also desire to place two result cells next to one another in order to compare and contrast description 394 and location 396.

A further alternative embodiment of the invention is illustrated in a second window 600 in FIG. 7. The second window 600 is similar to the first window 300 illustrated in FIG 5, but additionally includes a Boolean graphic 601 in the second workspace 636. The first window 300 and the second window 600 have identical menu bars 301. The first tool bar 303 of the first window 300 and the second tool bar 603 of the second window 600 both have a search by group

314 and an actions group 328, but the second tool bar 603 does not include the combine group 322 located in the first tool bar 303. The search cells do not include the negate icon 350. The search cells within the second window 600 do include an advanced button 351, which when selected causes additional features to be displayed within the search cell. A search cell 420 in which the advanced button 351 has been selected is illustrated in FIG. 10. When the advanced button 351 is selected, the search cell 420 is expanded and additional features including a facet display 400, a search button 402, a matching terms window 404, a select button 406, a cancel button 408 and a help button 410. The facet display 400 displays different fields for selection in which searches can be conducted. By selecting the facet field 400, a list of available fields pops up within the display providing the user the ability to select one or more of the fields. When the user selects the search button 402, the search, as specified within the search cell, is performed. When search terms are entered into the text field 348, additional potentially matching terms are displayed in the matching terms window 404. From this matching terms window 404, a user has the ability to select additional terms to be included in the text field 348. By selecting the advanced button 351 within the search cell 420 a second time, the search cell 420 will be minimized so that the additional features are hidden.

The Boolean graphic 601 is used to graphically display the hierarchical Boolean combination of search cells. The hierarchy of the Boolean graphic 601 reads from sub-equations on the left to progressively more senior equations to the right. Use of the Boolean graphic 601 avoids the need for hierarchical Boolean syntax, which is designated by the use of parenthesis, brackets, and curly braces. The Boolean graphic 601 illustrated in FIGS. 7-10 includes rectangular brackets. Within the Boolean graphic of the preferred embodiment of the present invention, curly braces are used to display the hierarchical Boolean combination of search cells.

In the second workspace 636, a Boolean combination of five search cells is displayed. The five search cells are labeled cell A, cell B, cell C, cell D, and cell E. To the right of the search cells are four rectangular brackets: D-1, D-2, D-3, and D-4. D-1 is to the right of cell B and cell C. The bracket D-2 is to the right of cell D and cell E. The bracket D-3 is to the right of the bracket D-1 and the bracket D-2. The bracket D-4 has two legs. The first leg of the bracket D-4 is to the right of cell A and above the bracket D-1 and the bracket D-3. The second leg of the bracket D-4 is to the right of the bracket D-3. Each rectangular bracket has a Boolean selector: the bracket D-1 has a first Boolean selector 618 labeled "OR"; the bracket D-2 has a second Boolean selector 620 labeled "AND"; the bracket D-3 has a third Boolean selector 622

labeled “AND”; and the bracket D-4 has a Boolean selector 624 labeled “AND”. Any of the Boolean selectors 618, 620, 622 and 624 can be toggled between “AND,” “OR” and “MINUS” by clicking on the Boolean selector with the computer mouse.

The Boolean graphic 601 also includes four pins: a pin AB 616 representing the relationship between the adjacent search cells A and B, a pin BC 610 representing the relationship between the adjacent search cells B and C, a pin CD 612 representing the relationship between the adjacent search cells C and D, and a pin DE 614 representing the relationship between the adjacent search cells D and E. A final Boolean equation summarizing the exemplary Boolean expression displayed in the second workspace 636 is represented by the Boolean expression $A[(B+C)(DE)]$ 626, in which an OR operation is performed on the search cells B and C, an AND operation is performed on the search cells D and E, the results of the two operations are ANDed together to form a first result, and an AND operation is performed on the first result and search cell A.

Creation of a Boolean equation using the Boolean graphic 601 is illustrated in FIGs. 8A to 8D. FIG. 8D represents the same Boolean equation found in the Boolean graphic 601 of FIG. 7 and includes cut-away views of the five search cells (cell A, cell B, cell C, cell D, and cell E) in a left column, rectangular brackets to the right of the five search cells, and the four pins (pin AB 616, pin BC 610, pin CD 612, and pin DE 614). Before the manipulation of any pins as illustrated in FIG. 8A, all four pins are adjacent to the right border of the search cells to which they correspond, the Boolean graphic 601 includes only a single rectangular bracket A-1 with Boolean selector A-1 519, and the Boolean equation is ABCDE 501, which performs an AND operation on all of the search cells A, B, C, D, and E. To move a pin, the searcher uses a drag and drop action with a cursor control device. FIG. 8B illustrates the appearance of the Boolean graphic 601 after the pin CD 612 has been moved to the right. The movement of pin CD 612 results in the generation of three rectangular brackets: the bracket B-1 with Boolean selector B-1 544, the bracket B-2 with Boolean selector B-2 546, and the bracket B-3 with Boolean selector B-3 548. The resulting Boolean equation is $(ABC)(DE)$ 521, which can be simplified as ABCDE and is still an AND operation performed on all of the search cells A, B, C, D, and E. When the Boolean selector C-1 574 is clicked, the Boolean logic of the Boolean graphic 601 becomes $(A+B+C)(DE)$ 551, as illustrated in FIG. 8C. FIG. 8C includes three rectangular brackets: the bracket C-1 with Boolean selector C-1 574, the bracket C-2 with Boolean selector C-2 576, and the bracket C-3 with Boolean selector C-3 578. When the pin AB 616 is moved to the right past

the pin CD 612 as illustrated in FIG. 8D, the Boolean graphic is altered such that four rectangular brackets are defined: the bracket D-1 with Boolean selector D-1 618, the bracket D-2 with Boolean selector D-2 620, the bracket D-3 with Boolean selector D-3 622, and the bracket D-4 with Boolean selector D-4 624. The resulting Boolean equation is $A[(B+C)(DE)]$ 581, in which an OR operation is performed on the search cells B and C, an AND operation is performed on the search cells D and E, the results of the two operations are ANDed together to form a first result, and an AND operation is performed on the first result and search cell A.

The Boolean graphic 601 can also be represented with the use of square brackets. FIGs. 8E to 8H illustrate the same four Boolean equations graphically represented by FIGs. 8A to 8D, respectively. Except for the use of square brackets instead of rectangular brackets, FIGs. 8E to 8H are identical to FIGs. 8A to 8D, respectively. FIGs. 8E to 8H include the cut-away views of the five search cells (cell A, cell B, cell C, cell D, and cell E) in a left column, square brackets to the right of the five search cells, and the four pins (pin AB 616, pin BC 610, pin CD 612, and pin DE 614). FIG. 8E includes only a single square bracket E-1 with Boolean selector E-1 712. FIG. 8F includes three square brackets: the bracket F-1 with Boolean selector F-1 736, the bracket F-2 with Boolean selector F-2 738, and the bracket F-3 with Boolean selector F-3 740. FIG. 8G includes three square brackets: the bracket G-1 with Boolean selector G-1 766, the bracket G-2 with Boolean selector G-2 768, and the bracket G-3 with Boolean selector G-3 770. FIG. 8H includes four square brackets: the bracket H-1 with Boolean selector H-1 786, the bracket H-2 with Boolean selector H-2 788, the bracket H-3 with Boolean selector H-3 798, and the bracket H-4 with Boolean selector H-4 799. Pins are moved and Boolean selectors are toggled in the same manner whether the Boolean graphic 601 employs rectangular brackets or square brackets. Square brackets can also be replaced by angle brackets, braces, or parenthesis.

A second alternate embodiment of the invention is illustrated in a third window 800 in FIG. 9. The third window 800 is similar to the first window 300 of FIG 5, but with the replacement of a Boolean folder tree 801 in the third workspace 802. The first window 300 and the third window 800 have identical menu bars 301 and first tool bar 303. The Boolean folder tree 801 is used to graphically display the hierarchical Boolean combination of search cells. The hierarchy of the Boolean folder tree 801 follows the general folder/sub-folder hierarchy of folder trees. As with use of the Boolean graphic 601 shown in FIG. 7, the Boolean folder tree 801 avoids the need for hierarchical Boolean syntax, which is designated by the use of parenthesis, brackets, and curly braces.

In the third workspace 802, a Boolean combination of five search cell icons is displayed. The five search cell icons are represented by small rectangles and labeled A, B, C, D, and E. The five search cell icons are arranged in a folder tree structure with four combination folders: a junior OR folder 804, a junior AND folder 806, a senior AND folder 808, and a top AND folder 810. The junior OR folder 804 includes the search cell icons B and C, as indicated by the first folder branch 805 connections. The junior AND folder 806 includes the search cell icon icons D and E, as indicated by the second folder branch 807. The senior AND folder 808 includes the junior OR folder 804 and the junior AND folder 806, as indicated by the third folder branch 809. The top AND folder 810 includes the search cell icon A and the senior AND folder 808, as indicated by the fourth folder branch 811.

A query is created in the third window 800, starting with a blank third workspace 802. The blank third workspace 802 is created using the "New" pull-down option from the session menu 302 (and also when the first search window 300 is first opened). Search cells identical to those shown in the first workspace 336 of FIG. 5 are added to the third workspace 802 one at a time in the same manner they are added to the first workspace 336. Once created, search cells located in the third workspace 802 can be reduced into the small, rectangular search cell icons shown in FIG. 9 by double clicking a given search cell. In corollary, double clicking search cell icons will expand the search cell icon into a complete search cell.

Search cell icons can be combined into a Boolean query using combination folders by first flagging specific search cell icons, then selecting the logical AND button 324, the logical OR button 326 or the logical MINUS button 327. Selecting the logical AND button 324 creates a combination folder labeled "AND," selecting the logical OR button 326 creates a combination folder labeled "OR" and selecting the logical MINUS button 327 creates a combination folder labeled "MINUS." Details of a combination folder can be reviewed by double clicking the specific combination folder, expanding the folder into a combination cell 344 as shown in FIG. 5 or selecting the combination cell and then clicking the expand/collapse button 333. Combinations cells, in corollary, can be reduced to combination folders by double clicking or selecting the combination cell and then clicking the expand/collapse button 333. Search cell icons and combination folders can be dragged and dropped within the third workspace 802 to alter the Boolean query. Search cell icons and combination folders can also be deleted by flagging the items to be deleted then selecting the remove button 330. As in the first workspace, the third workspace 802 treats queries as a file. Queries within the third workspace 802 can be

saved, closed, opened, and merged within other queries. Along with the query, a user can also save a comments dialog, allowing the user to save notes and thoughts about the search query. Queries can also be executed in the third workspace 802 by selecting the results button 322, which will change the appearance of the workspace 802 as shown in FIG. 6.

5 Due to extensive searcher interaction with the graphical user interface, local execution of the applet on the client increases the effectiveness of the invention. An applet is a small Java software program that can be run by an internet browser on multiple computer platforms. Applets can be permanently saved on the hard drive of the client computer as a browser plugin and executed as a resident applet. Applets can also be downloaded and executed as a server
10 transferred applet that is not saved within the browser. Download time for the applet is negligible because the applet is not a stand alone program but rather an augmentation of the client's browser. The size of the applet can be as small as 500 kilobytes. If the applet were not executed on the client, each page refresh would have to be requested from the server over a communication line, assembled by the server, and sent back to the client over a communication
15 line, resulting in significant time delays with each interaction between the searcher and the invention's graphical user interface. With the use of the applet, the searcher can insert cells, move pins, and enter text in real time. The applet also allows the searcher to continue with their modification of the graphical user interface while queries are delivered to the server and search results are sent back to the client. Using an applet, the searcher commands the resources of a
20 remote server with the speed of a local client.

The local client computer can be a computer system with a graphical user interface, such as the exemplary computer system illustrated in FIG. 1. Resident applets can be saved on the mass storage device 406 and moved to the main memory 402 during execution. The search engine's graphical user interface of the present invention can be displayed on the CRT 410 and
25 controlled by use of the computer mouse 405 and the keyboard 404.

The present invention is well designed for use by the general population. First, the invention is designed for use with a standard internet browser. Proprietary software does not have to be installed by the searcher. Second, use of the applet permits the local client to generate many of the internet browser screen refreshes. Thus, unlike a typical session with an
30 internet browser, the searcher need not wait for a new screen to be requested and received over a communication line each time a button on the screen is pressed. Unhindered by communication delays, the searcher has the ability to manipulate the Boolean graphics and the search cells in real

time. Local screen refreshes also make the use of user-friendly graphics feasible, such that novice searchers are able to draft and execute well-structured Boolean queries.

The present invention has been described in terms of specific embodiments incorporating details to facilitate the understanding of principles of construction and operation of the invention.

5 Such reference herein to specific embodiments and details thereof is not intended to limit the scope of the claims appended hereto. It will be apparent to those skilled in the art that modifications may be made in the embodiment chosen for illustration without departing from the spirit and scope of the invention.

10

CLAIMS

We claim:

- 1 1. A graphical user interface for building Boolean queries and viewing search results
2 comprising:
3 a. a workspace capable of enclosing one or more search cells and one or more
4 combination cells;
5 b. a means for inserting search cells into the workspace, each search cell having a
6 search by type icon, a text field and a hit count;
7 c. a means for inserting combination cells into the workspace each combination cell
8 having a combine icon, a hit count and an expression field, wherein content of the
9 expression field within a selected combination cell is determined by the combine
10 icon and selection of search cells to be included within the selected combination
11 cell;
12 d. a means for deleting selected search cells and combination cells from the
13 workspace; and
14 e. a means for viewing search results of selected search cells and flagged
15 combination cells.
- 1 2. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein the search cells and the combination cells are capable of being
3 graphically relocated within the workspace.
- 1 3. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 2, wherein:
3 a. an applet running on a client controls insertion, selection, movement, and deletion
4 of the search cells and the combination cells; and
5 b. a server:
6 i. executes queries received over an internet connection from the client; and
7 ii. delivers search results back to the client over the internet.

- 1 4. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 3, wherein the client is local and the server is remote.
- 1 5. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein the search by type icon is selected from a group including a category
3 icon, a keyword icon, and a media type icon.
- 1 6. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein the combine icon is selected from a group including a logical AND icon,
3 a logical OR icon and a logical MINUS icon.
- 1 7. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein content of the text field is capable of being selected using a category
3 walker.
- 1 8. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein information displayed by the means for viewing search results includes
3 a title, a media type, and a location for each search result.
- 1 9. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 8, wherein the location is a URL address.
- 1 10. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein search parameters displayed in the workspace are treated as a session.
- 1 11. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 10, wherein a user has ability to save, open, import, and close the session.
- 1 12. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 1, wherein the search results are treated as a collection.

13. The graphical user interface for building Boolean queries and viewing search results as in claim 12, wherein a user has ability to save, open, import, and close the collection.

14. A method of building Boolean queries and viewing search results using a graphical user interface comprising the steps of:

- a. opening a window including a workspace and a toolbar;
- b. inserting two or more search cells into the workspace, each search cell having a search by type icon, a text field and a hit count; and
- c. entering text in the text fields of the search cells;
- d. selecting two or more search cells;
- e. inserting a combination cell into the workspace, the combination cell having a combine icon, a hit count and an expression field, wherein content of the expression field is determined by the combine icon and selection of search cells to be included within the combination cell;
- f. selecting the combination cell;
- g. executing the query represented by each selected combination cell; and
- h. viewing search results in the workspace.

15. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 14, wherein the search cells and the combination cells are capable of being graphically relocated within the workspace.

16. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 15, wherein:

- a. an applet running on a client controls insertion, selection, and movement of the search cells and the combination cells; and
- b. a server:
 - i. executes queries received over an internet connection from the client; and
 - ii. delivers search results back to the client over the internet.

17. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 14, wherein the client is local and the server is remote.

- 1 18. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, wherein the search by type icon is selected from a group
3 including category, keyword and media type.
- 1 19. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, wherein the combine icon is selected from a group including
3 a logical AND icon, a logical OR icon and a logical MINUS icon.
- 1 20. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, wherein content of the text field is capable of being selected
3 using a category walker.
- 1 21. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, further comprising the step of displaying each search result,
3 including a title, a media type, and a location.
- 1 22. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 21, wherein the location is a URL address.
- 1 23. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, wherein search parameters displayed in the workspace are
3 treated as a session.
- 1 24. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 23, wherein a user has ability to save, open, import, and close
3 the session.
- 1 25. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 14, wherein the search results are treated as a collection.

- 1 26. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 25, wherein a user has ability to save, open, import, and close
3 the collection.
- 1 27. A graphical user interface for building Boolean queries and viewing search results
2 comprising:
3 a. a window including a workspace and a tool bar;
4 b. one or more search cells located within the workspace, each search cell having a
5 search by type icon, a text field and a hit count; and
6 c. one or more combination cells, each combination cell having a combine icon, a hit
7 count and an expression field, wherein content of the expression field within a
8 selected combination cell is determined by the combine icon and selection of
9 search cells to be included within the selected combination cell.
- 1 28. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 27, wherein the search cells and the combination cells are capable of being
3 graphically relocated within the workspace.
- 1 29. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 28, wherein:
3 a. an applet running on a client controls insertion, selection, movement, and deletion
4 of the search cells and the combination cells; and
5 b. a server executes queries received over an internet connection from the client and
6 delivers search results back to the client over the internet.
- 1 30. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 29 wherein the client is local and the server is remote.
- 1 31. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 27, wherein the search by type icon is selected from a group including a category
3 icon, a keyword icon, and a media type icon.

32. The graphical user interface for building Boolean queries and viewing search results as in claim 27, wherein the combine icon is selected from a group including a logical AND icon, a logical OR icon and a logical MINUS icon.

33. The graphical user interface for building Boolean queries and viewing search results as in claim 27, wherein content of the text field is capable of being selected using a category walker.

34. The graphical user interface for building Boolean queries and viewing search results as in claim 27, wherein the toolbar includes a search by group, a combine group, and an actions group:

- a. the search by group having a category button, a keyword button, and a date button, wherein:
 - i. when the category button is selected, one search cell is generated with a category icon as the search by icon;
 - ii. when the keyword button is selected, one search cell is generated with a keyword icon as the search by icon; and
 - iii. when the date button is selected, one search cell is generated with a date icon as the search by icon;
- b. the combine group having a logical AND button, a logical OR button and a logical MINUS button, wherein:
 - i. when the logical AND button is selected, one combination cell is generated with a logical AND icon as the combine icon;
 - ii. when the logical OR button is selected, one combination cell is generated with a logical OR icon as the combine icon; and
 - iii. when the logical MINUS button is selected, one combination cell is generated with a logical MINUS icon as the combine icon; and
- c. the actions group having a remove button, a results button, an expand/collapse button and a quit button, wherein:
 - i. when the remove button is selected, selected combination cells and selected search cells are removed from the workspace;

- ii. when the results button is selected, a query defined by each selected combination cell is executed and a results tab is generated;
- iii. when the expand/collapse button is selected, the selected combination cell is alternately collapsed into itself and expanded to show search cells within the selected combination cell; and
- iv. when the quit button is selected, the window is closed.

35. The graphical user interface for building Boolean queries and viewing search results as in claim 34, wherein when the results button is selected, the search results are displayed within the workspace.

36. The graphical user interface for building Boolean queries and viewing search results as in claim 35, wherein the workspace includes a title, a media type, and a location for each displayed search result.

37. The graphical user interface for building Boolean queries and viewing search results as in claim 36, wherein the location is a URL address.

38. The graphical user interface for building Boolean queries and viewing search results as in claim 34, wherein the window also includes a menu bar.

39. The graphical user interface for building Boolean queries and viewing search results as in claim 38, wherein the menu bar includes a session menu, a collection menu, a query menu, a select menu, and a help menu, wherein:

- a. the session menu includes a new session option, an open session option, a save session option, a close session option, and an exit option;
- b. the collection menu includes a new collection option, an open collection option, a merge collection option, a save option, and a close collection option;
- c. the query menu includes a search by category option, a search by keyword option, a search by numeric and date option, a combine by logical AND option, a combine by logical OR option, a combine by logical MINUS option and an execute search option;

- d. the select menu includes a select all option, an unselect all option, and a selection option; and
- e. the help menu includes an index option and an about option.

40. A graphical user interface for building Boolean queries and viewing search results comprising:

- a. a workspace capable of enclosing one or more search folders and one or more combination folders in a folder tree hierarchy;
- b. a means for inserting search folders into the workspace, each search folder configured for displaying a search by type icon, a text field, and a hit count;
- c. a means for inserting combination folders into the workspace, each combination folder having a combine icon;
- d. a means for moving selected search folders and selected combination folders within the folder tree hierarchy;
- e. a means for deleting selected search folders and selected combination folders from the workspace; and
- f. a means for viewing search results of selected search folders and selected combination folders.

41. A graphical user interface for building Boolean queries and viewing search results as in claim 40, wherein:

- a. an applet running on a client controls insertion, selection, movement, and deletion of the search folders and the combination folders; and
- b. a server executes queries received over an internet connection from the client and delivers search results back to the client over the internet.

42. A graphical user interface for building Boolean queries and viewing search results as in claim 41 wherein the client is local and the server is remote.

43. The graphical user interface for building Boolean queries and viewing search results as in claim 40, wherein the combine icon is selected from a group including a logical AND icon, a logical OR icon and a logical MINUS icon.

- 1 44. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 40, wherein content of the text field is capable of being selected using a category
3 walker.
- 1 45. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 40, wherein information displayed by the means for viewing includes a title, a
3 media type, and a location for each search result.
- 1 46. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 45, wherein the location is a URL address.
- 1 47. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 40, wherein search parameters displayed in the workspace are treated as a session.
- 1 48. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 47, wherein a user has ability to save, open and close the session.
- 1 49. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 40, wherein the search results are treated as a collection.
- 1 50. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 49, wherein a user has ability to save, open and close the collection.
- 1 51. A method of building Boolean queries and viewing search results using a graphical user
2 interface comprising the steps of:
3 a. opening a window including a workspace and a toolbar;
4 b. inserting two or more search folders into the workspace, each search folder
5 configured for displaying a search by type icon, a text field, and a hit count;
6 c. entering text in the text fields of the search folders;
7 d. selecting two or more search folders;

- e. inserting one or more combination folders into the workspace, each combination folder having a combine icon;
- f. arranging the search folders and combination folders within a folder tree hierarchy;
- g. selecting one combination folder thereby forming a selected combination folder;
- h. executing a query represented by each the selected combination folder; and
- i. viewing search results in the workspace.

52. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 51, wherein:

- a. an applet running on a client controls insertion, selection, movement, and deletion of the search folders and the combination folders; and
- b. a server executes queries received over an internet connection from the client and delivers search results back to the client over the internet.

53. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 52, wherein the client is local and the server is remote.

54. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 51, wherein the combine icon is selected from a group including a logical AND icon, a logical OR icon and a logical MINUS icon.

55. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 51 wherein content of the text field is capable of being selected using a category walker.

56. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 51 further comprising the step of displaying each search result, including a title, a media type, and a location.

57. The method for building Boolean queries and viewing search results using a graphical user interface as in claim 56 wherein the location is a URL address.

1 58. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 51, wherein search parameters displayed in the workspace are
3 treated as a session.

1 59. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 51 wherein the search results are treated as a collection.

1 60. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 59, wherein a user has ability to save, open and close the
3 collection.

1 61. A graphical user interface for building Boolean queries and viewing search results
2 comprising:
3 a. a window including a workspace, a tool bar, and a folder tree hierarchy;
4 b. two or more search folders located within the folder tree hierarchy, each search
5 folder capable of being moved within the folder tree hierarchy and configured for
6 displaying a search by type icon, a text field, and a hit count;
7 c. one or more combination folders, each combination folder capable of being
8 moved within the folder tree hierarchy and having a combine icon.

1 62. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 61, wherein:
3 a. an applet running on a client controls insertion, selection, movement, and deletion
4 of the search folders and the combination folders; and
5 b. a server executes queries received over an internet connection from the client and
6 delivers search results back to the client over the internet.

1 63. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 61, wherein the combine icon is selected from a group including a logical AND
3 icon, a logical OR icon and a logical MINUS icon.

64. The graphical user interface for building Boolean queries and viewing search results as in claim 61, wherein content of the text field is capable of being selected using a category walker.

65. The graphical user interface for building Boolean queries and viewing search results as in claim 61, wherein the toolbar includes a search by group, a combine group, and an actions group:

- a. the search by group having a category button, a keyword button, and a date button, wherein:
 - i. when the category button is selected, one search folder is generated with a category icon as the search by icon;
 - ii. when the keyword button is selected, one search folder is generated with a keyword icon as the search by icon; and
 - iii. when the date button is selected, one search folder is generated with a date icon as the search by icon;
- b. the combine group having a logical AND button, a logical OR button and a logical MINUS button, wherein:
 - i. when the logical AND button is selected, one combination folder is generated with a logical AND icon as the combine icon;
 - ii. when the logical OR button is selected, one combination folder is generated with a logical OR icon as the combine icon; and
 - iii. when the logical MINUS button is selected, one combination folder is generated with a logical MINUS icon as the combine icon; and
- c. the actions group having a remove button, a results button, an expand/collapse button and a quit button, wherein:
 - i. when the remove button is selected, selected combination folders and selected search folders are removed from the workspace;
 - ii. when the results button is selected, a query defined by each selected combination folder is executed and a results tab is generated;
 - iii. when the expand/collapse button is selected, the selected combination cell is alternately collapsed into itself and expanded to show search cells within the selected combination cell; and

iv. when the quit button is selected, the window is closed.

66. The graphical user interface for building Boolean queries and viewing search results as in claim 65, wherein when the results button is selected, the search results are displayed within the workspace.

67. The graphical user interface for building Boolean queries and viewing search results as in claim 66, wherein the workspace includes a title, a media type, and a location for each displayed search result.

68. The graphical user interface for building Boolean queries and viewing search results as in claim 67, wherein the location is a URL address.

69. The graphical user interface for building Boolean queries and viewing search results as in claim 65, wherein the window also includes a menu bar.

70. The graphical user interface for building Boolean queries and viewing search results as in claim 69, wherein the menu bar includes a session menu, a collection menu, a query menu, a select menu, and a help menu, wherein:

- a. the session menu includes a new session option, an open session option, a save session option, a close session option, and an exit option;
- b. the collection menu includes a new collection option, an open collection option, a merge collection option, a save option, and a close collection option;
- c. the query menu includes a search by category option, a search by keyword option, a search by numeric and date option, a combine by logical AND option, a combine by logical OR option, a combine by logical MINUS option and an execute search option;
- d. the select menu includes a select all option, an unselect all option, and a selection option; and
- e. the help menu includes an index option and an about option.

- 1 71. A graphical user interface for building Boolean queries and viewing search results
2 comprising:
- 3 a. a workspace capable of enclosing:
- 4 i. search cells stacked in a first column, each search cell having a search by
5 type icon, a text field and a hit count;
- 6 ii. one or more pins graphically displayed between each pair of adjacent
7 search cells, each pin having a shaft and a pinhead; and
- 8 iii. one or more combination brackets arranged in a Boolean hierarchy within
9 a second column adjacent to the first column, each combination bracket
10 displayed adjacent to at least two search cells and having a combine icon
11 and a combination bracket flag; and
- 12 b. means for inserting search cells into the workspace;
- 13 c. means for graphically sliding pins horizontally within the second column:
- 14 i. to generate and manipulate combination brackets; and
- 15 ii. to alter Boolean hierarchy of the combination brackets;
- 16 d. means for moving selected search cells;
- 17 e. means for deleting selected search cells from the workspace; and
- 18 f. means for viewing search results corresponding to a Boolean query.
- 19 72. The graphical user interface for building Boolean queries and viewing search results as in
20 claim 71, wherein:
- 21 a. an applet running on a client controls:
- 22 i. insertion, selection, movement, and deletion of the search cells; and
- 23 ii. the generation and manipulation of the combination brackets; and
- 24 b. a server:
- 25 i. executes queries received over an internet connection from the client; and
- 26 ii. delivers search results back to the client over the internet.
- 27 73. The graphical user interface for building Boolean queries and viewing search results as in
28 claim 71 wherein the client is local and the server is remote.

- 1 74. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein the combine icon is selected from a group including a logical AND
3 icon, a logical OR icon and a logical MINUS icon.
- 1 75. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein content of the text field is capable of being selected using a category
3 walker.
- 1 76. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein information displayed by the means for viewing includes a title and a
3 location for each search result.
- 1 77. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 76, wherein the location is a URL address.
- 1 78. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein search parameters displayed in the workspace are treated as a session.
- 1 79. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 78, wherein a user has ability to save, open and close the session.
- 1 80. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein the search results are treated as a collection.
- 1 81. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 80, wherein a user has ability to save, open and close the collection.
- 1 82. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 71, wherein graphical representation of the combination bracket is selected from a
3 group including a rectangular bracket, a square bracket and a curly brace.

1 83. A method for building Boolean queries and viewing search results using a graphical user
2 interface comprising the steps of:

- 3 a. opening a window including a workspace and a toolbar;
- 4 b. stacking two or more search cells into a first column of the workspace:
 - 5 i. each search cell having a search by type icon, a text field and a hit count;
 - 6 and
 - 7 ii. each pair of adjacent search cells having a pin graphically displayed
 - 8 therebetween, each pin having a shaft and a pinhead; and
- 9 c. entering text in the text fields of the search cells;
- 10 d. graphically sliding pins horizontally within a second column adjacent to the first
11 column:
 - 12 i. to generate and manipulate a plurality of combination brackets, each
 - 13 combination bracket adjacent to at least two search cells and each
 - 14 combination bracket having a combine icon; and
 - 15 ii. to alter a Boolean hierarchy of the combination brackets;
- 16 e. arranging placement of search cells and combination brackets to generate the
17 Boolean query;
- 18 f. executing the Boolean query represented within the workspace; and
- 19 g. viewing search results in the workspace.

1 84. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein the search by type icon is selected from a group
3 including a category icon, a keyword icon, and a media type icon.

1 85. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein:

- 3 a. an applet running on a client controls:
 - 4 i. insertion, selection, movement, and deletion of the search cells; and
 - 5 ii. the generation and manipulation of the combination brackets; and
- 6 b. a server:
 - 7 i. executes queries received over an internet connection from the client; and
 - 8 ii. delivers search results back to the client over the internet.

1 86. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein the combine icon is selected from a group including
3 a logical AND icon, a logical OR icon and a logical MINUS icon.

1 87. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein content of the text field is capable of being selected
3 using a category walker.

1 88. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein information displayed by the view of each search
3 result includes a title and a location.

1 89. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 88, wherein the location is a URL address.

1 90. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein search parameters displayed in the workspace are
3 treated as a session.

1 91. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 90, wherein a user has ability to save, open and close the
3 session.

1 92. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein the search results are treated as a collection.

1 93. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 92, wherein a user has ability to save, open and close the
3 collection.

1 94. The method for building Boolean queries and viewing search results using a graphical
2 user interface as in claim 83, wherein graphical representation of the combination bracket
3 is selected from a group including a rectangular bracket, a square bracket, and a curly
4 brace.

1 95. A graphical user interface for building Boolean queries and viewing search results
2 comprising:
3 a. a window including a workspace and a tool bar;
4 b. two or more search cells stacked in a first column of the workspace:
5 i. each search cell having a search by type icon, a text field and a hit count;
6 and
7 ii. each search cell movable within the first column;
8 c. a pin between each pair of adjacent search cells, each pin having a shaft and a
9 pinhead and each pin configured for horizontal movement within a second column
10 adjacent to the first column; and
11 d. one or more combination brackets located in the second column, each
12 combination bracket adjacent to at least two search cells and each combination
13 bracket having a combine icon, wherein movement of the pin head generates and
14 manipulates the combination brackets.

1 96. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein the search by icon is selected from a group including a category icon, a
3 keyword icon, and a media type icon.

1 97. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein each search cell includes an advanced button which when selected
3 causes the search cell to expand and additional features including facet type and matching
4 terms to be displayed within the search cell.

1 98. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein:

- 3 a. an applet running on a client controls:
- 4 i. insertion, selection, movement, and deletion of the search cells; and
 - 5 ii. the generation and manipulation of the combination brackets; and
- 6 b. a server:
- 7 i. executes queries received over an internet connection from the client; and
 - 8 ii. delivers search results back to the client over the internet.

1 99. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein the combine icon is selected from a group including a logical AND
3 icon, an logical OR icon and a logical MINUS icon.

1 100. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein content of the text field is capable of being selected with using a
3 category walker.

1 101. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein the toolbar includes a search by group, a combine group, and an
3 actions group:

- 4 a. the search by group having a category button, a keyword button, and a date
5 button, wherein:
- 6 i. when the category button is selected, one search cell is generated with a
7 category icon as the search by icon;
 - 8 ii. when the keyword button is selected, one search cell is generated with a
9 keyword icon as the search by icon; and
 - 10 iii. when the date button is selected, one search cell is generated with a date
11 icon as the search by icon;
- 12 b. the combine group having a logical AND button, a logical OR button and a
13 logical MINUS button, wherein:
- 14 i. the logical AND button is selected, one combination folder is generated
15 with a logical AND icon as the combine icon;

- 16 ii. when the logical OR button is selected, one combination folder is
17 generated with a logical OR icon as the combine icon; and
18 iii. when the logical MINUS button is selected, one combination folder is
19 generated with a logical MINUS icon as the combine icon; and
20 c. the actions group having a remove button, a results button, an expand/collapse
21 button and a quit button, wherein:
22 i. when the remove button is selected, selected combination folders and
23 selected search cells are removed from the workspace;
24 ii. when the results button is selected, a query defined by each selected
25 combination folder is executed and a results tab is generated; and
26 iii. when the expand/collapse button is selected, the selected combination cell
27 is alternately collapsed into itself and expanded to show search cells
28 within the selected combination cell; and
29 iv. when the quit button is selected, the window is closed.
- 1 102. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 101, wherein when the results button is selected, search results are displayed within
3 the workspace.
- 1 103. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 102, wherein the workspace includes a title and a location for each displayed search
3 result.
- 1 104. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 103, wherein the location is a URL address.
- 1 105. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 101, wherein the window also includes a menu bar.

- 1 106. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 105, wherein the menu bar includes a session menu, a collection menu, a query
3 menu, a select menu, and a help menu, wherein:
4 a. the session menu includes a new session option, an open session option, a save
5 session option, a close session option, and an exit option;
6 b. the collection menu includes a new collection option, an open collection option, a
7 merge collection option, a save option, and a close collection option;
8 c. the query menu includes a search by category option, a search by keyword option,
9 a search by numeric and date option, a combine by logical AND option, a
10 combine by logical OR option, a combine by logical MINUS option and an
11 execute search option;
12 d. the select menu includes a select all option, an unselect all option, and a selection
13 option; and
14 e. the help menu includes an index option and an about option.
- 1 107. The graphical user interface for building Boolean queries and viewing search results as in
2 claim 95, wherein graphical representation of the combination bracket is selected from a
3 group including a rectangular bracket, a square bracket, and a curly brace.

ABSTRACT

A graphical user interface is for building Boolean queries and viewing search results using search cells, combination cells, a toolbar, and a menu bar. The search cells are movable within a workspace and preferably contain a text field, a facet type designation, an advanced button and a hit count. Search cells can search records by a category, keyword, date or numeric value. Categories are preferably chosen with the assistance of a category walker. The combination cells are movable within the workspace and preferably contain a combine icon, a hit count and an expression field. The combination cell is a Boolean combination of two or more selected search cells. In the preferred embodiment of the invention, the Boolean logic is graphically depicted with the use of combination brackets. The contents of each combination bracket is controllable by the use of movable graphical pins that are located between each pair of search cells. The combination bracket can be represented by use of a rectangular bracket, a square bracket, or a curly brace. In an alternative embodiment of the invention, the Boolean query is depicted in a folder tree hierarchy. Search folders and combination folders are used to depict search cells and combination cells, respectively. Search folders contain the same information as search cells, but search folders can be toggled to appear as small folder icons within the workspace. Combination folders are represented as small folder icons containing a logical AND icon, a logical OR icon or a logical MINUS icon. Search folders are shown to be included within combination folders through the use of the folder tree hierarchy, just as computer files are shown to be included within a folder in a typical file manager program. In the preferred embodiment, the invention runs an internet browser and is coupled to a remote server through the internet. To minimize communication delays across the internet, an applet can be run on the local client to perform most graphical display manipulations prompted by the searches manipulation of the search cells, the combination cells, the toolbar, and the menu. The remote server is used to execute queries received over the internet from the local client and deliver search results back to the local client over the internet. Optionally, the client can take on some or all of the tasks of the server in executing queries over the internet.

FIG. 1 - PRIOR ART

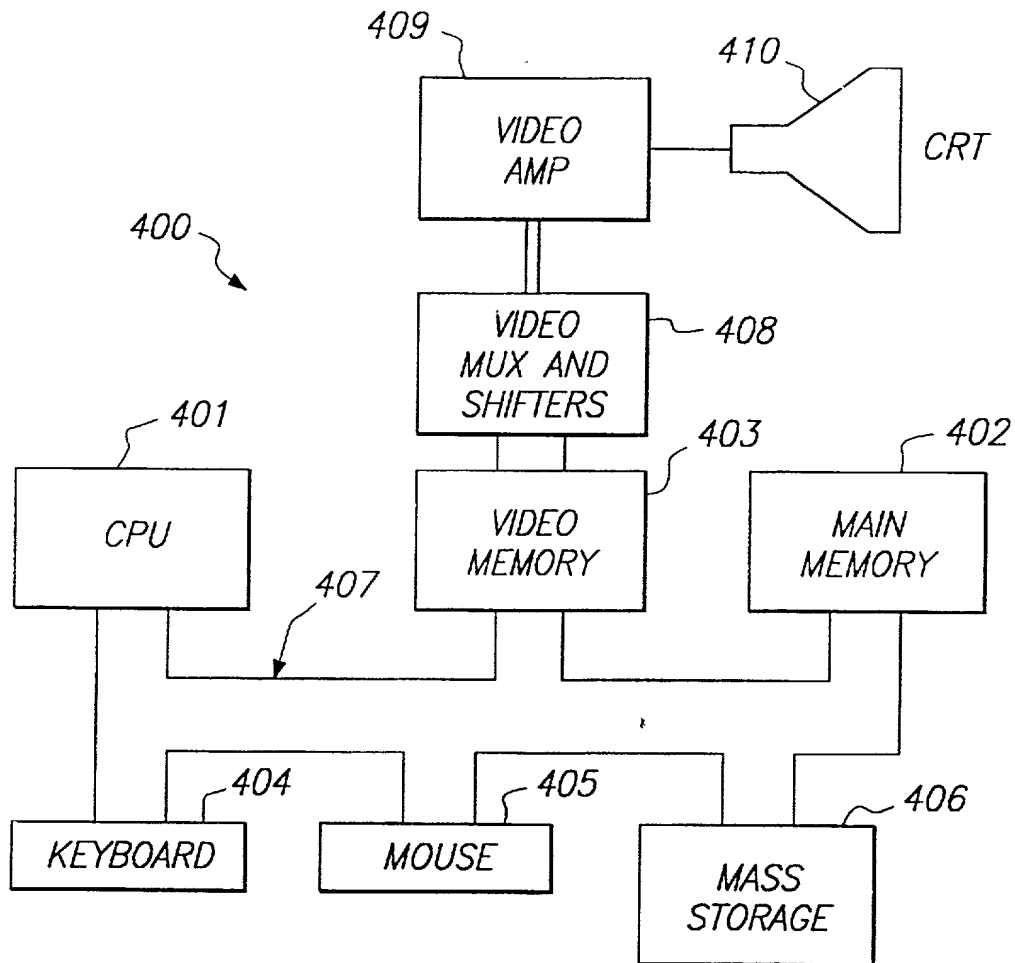


FIG. 2 - PRIOR ART

100 ↙

CLIENT

102→

CATEGORY

104→

106→

CATEGORY WALKER

QUERY

108→

SEGMENTS

110↵

BOOLEAN OPERATORS

112↵

SEARCH

114↵

DATE:

116↵

118↵

120↵

122↵

SAVE QUERY

124↵

CLOSE

126↵

HELP

128↵

FIG. 3 - PRIOR ART

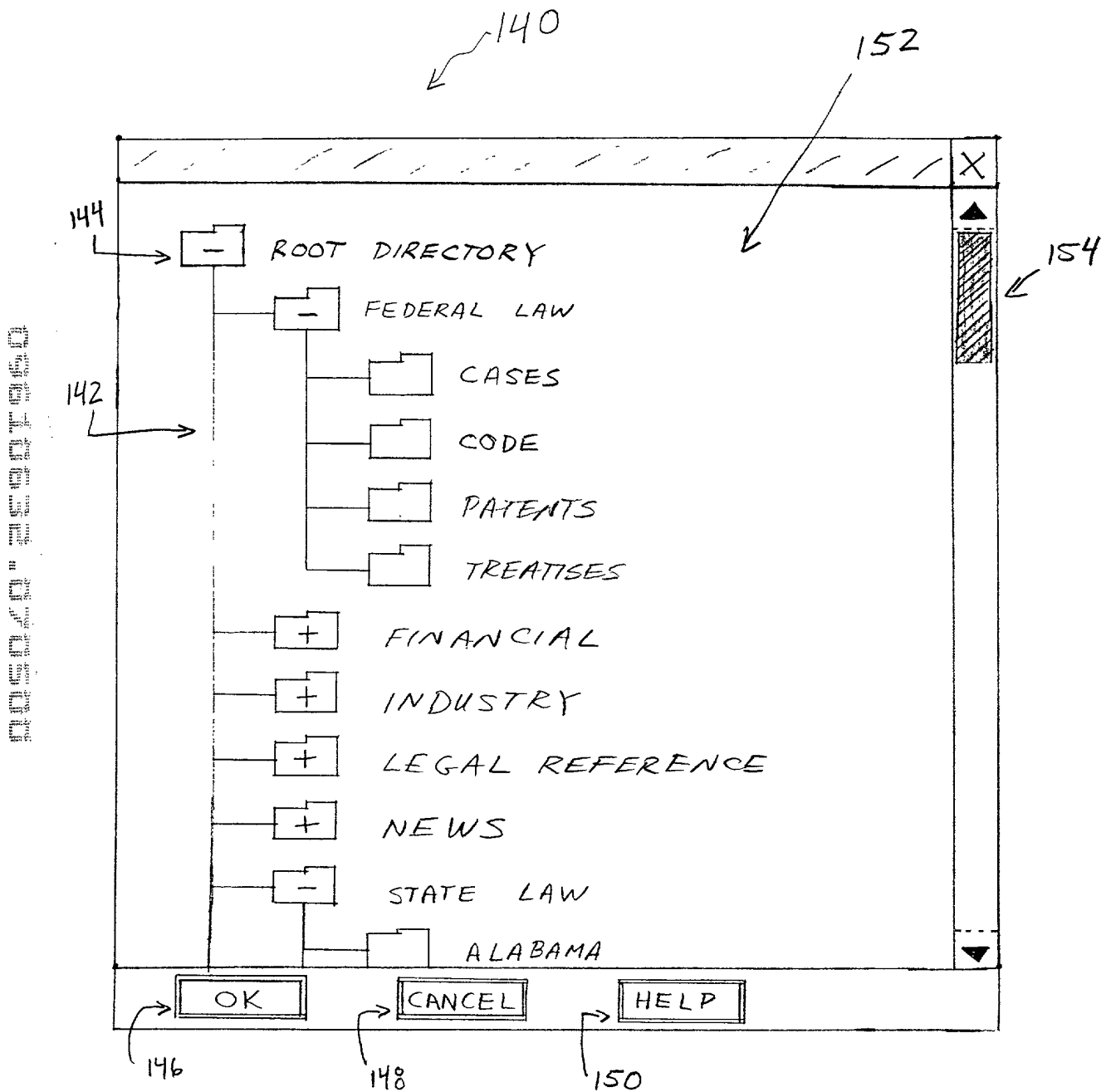
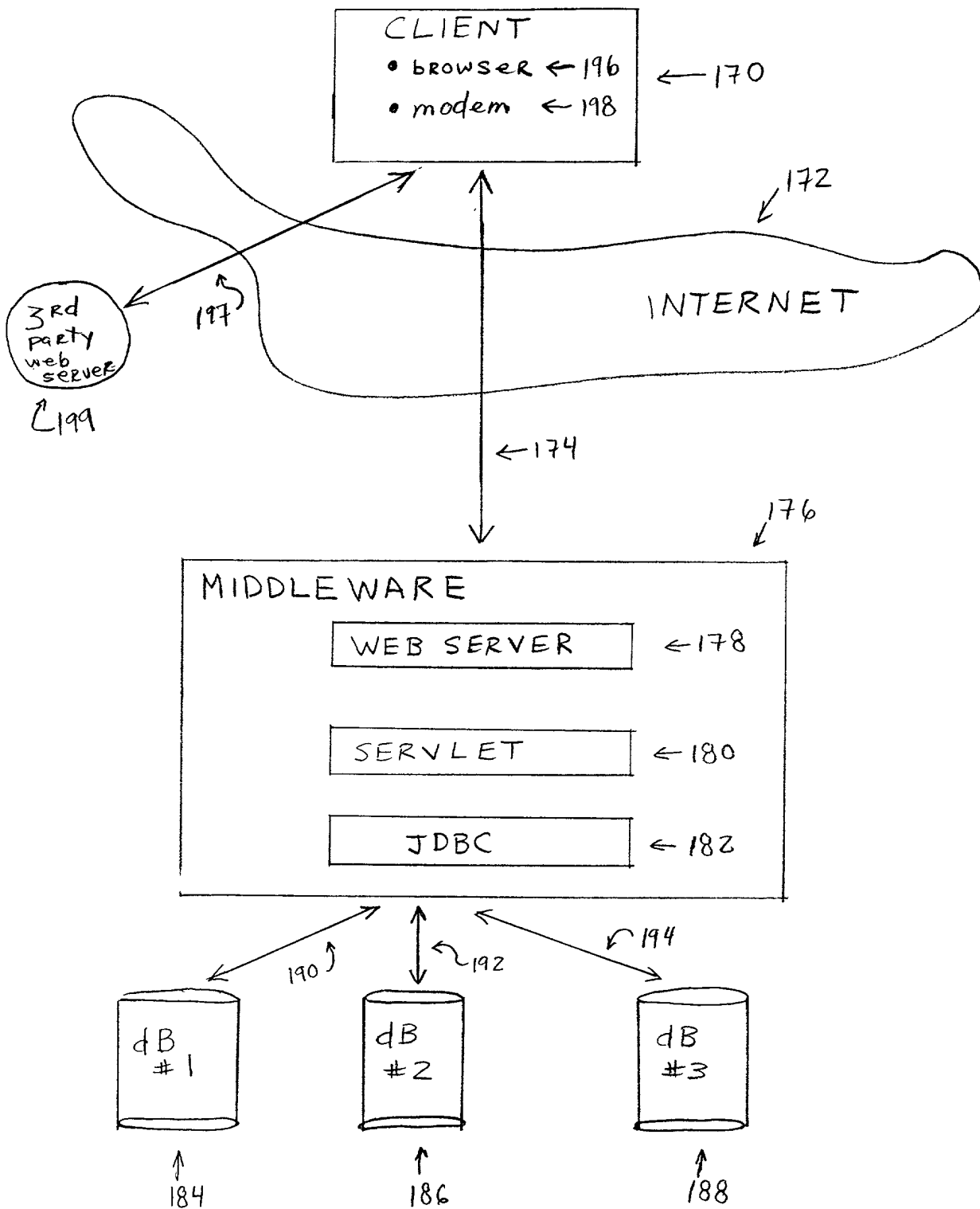


FIG. 4 — PRIOR ART



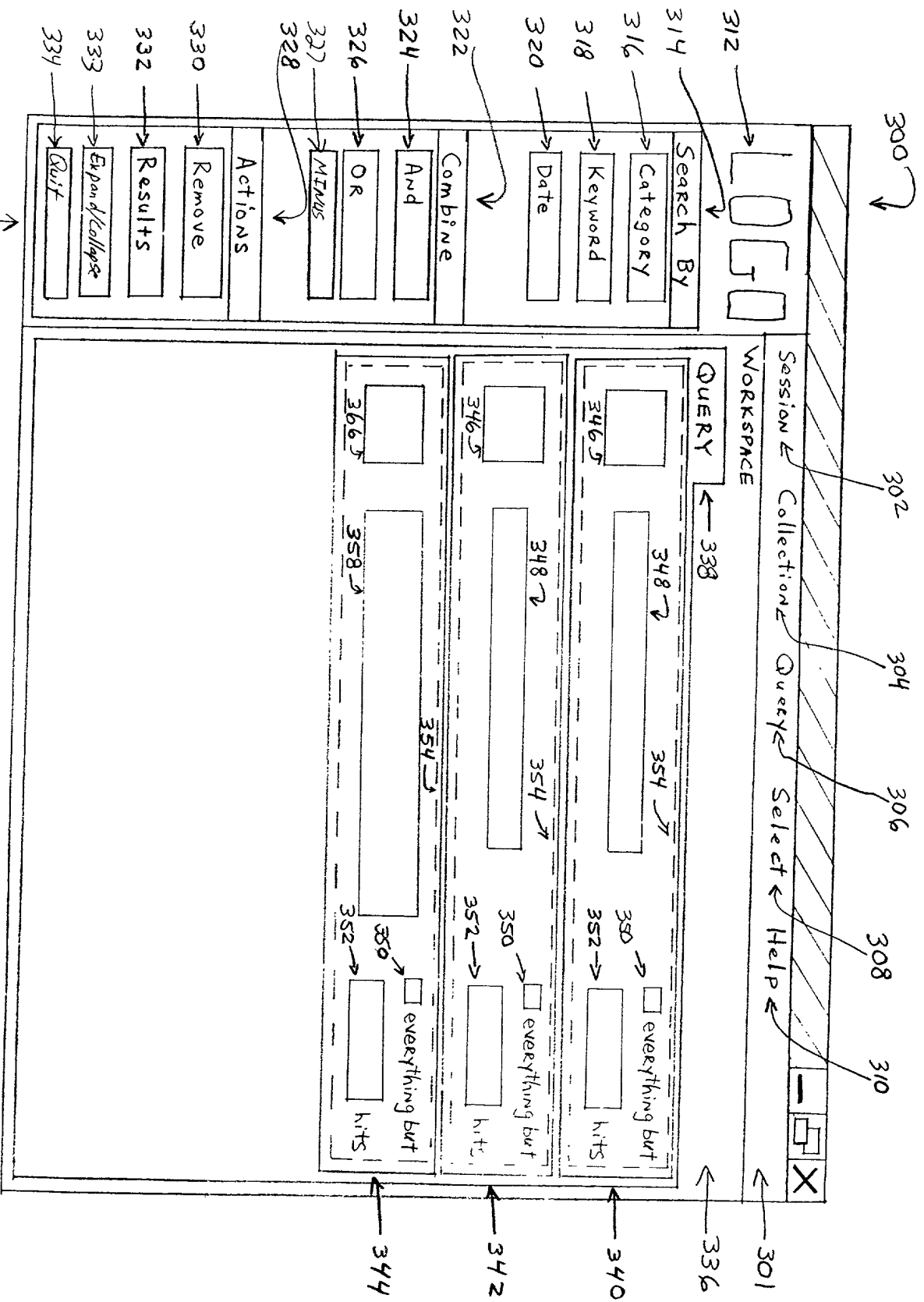


FIG. 5

FIG. 5 is a block diagram of a search interface.

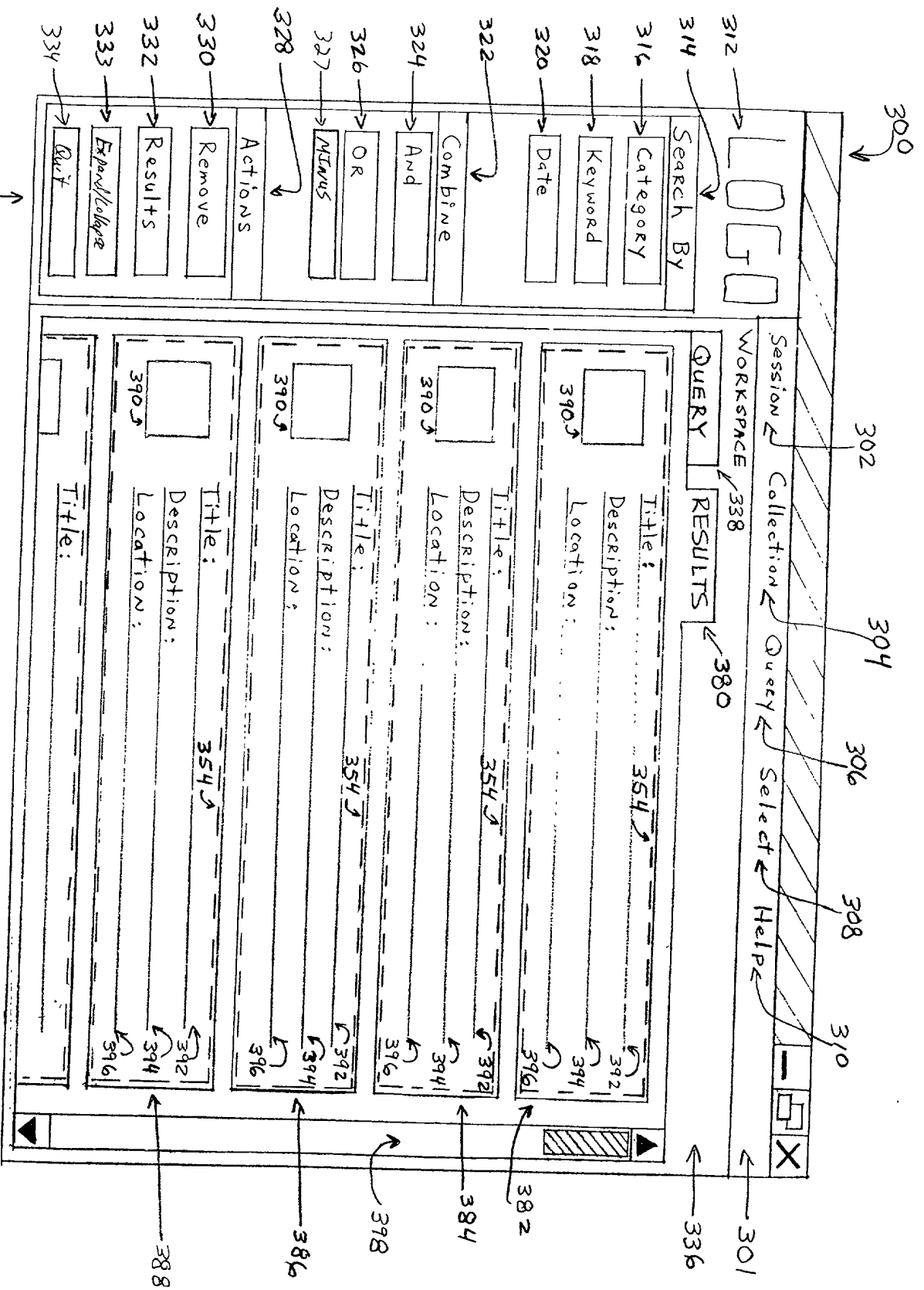
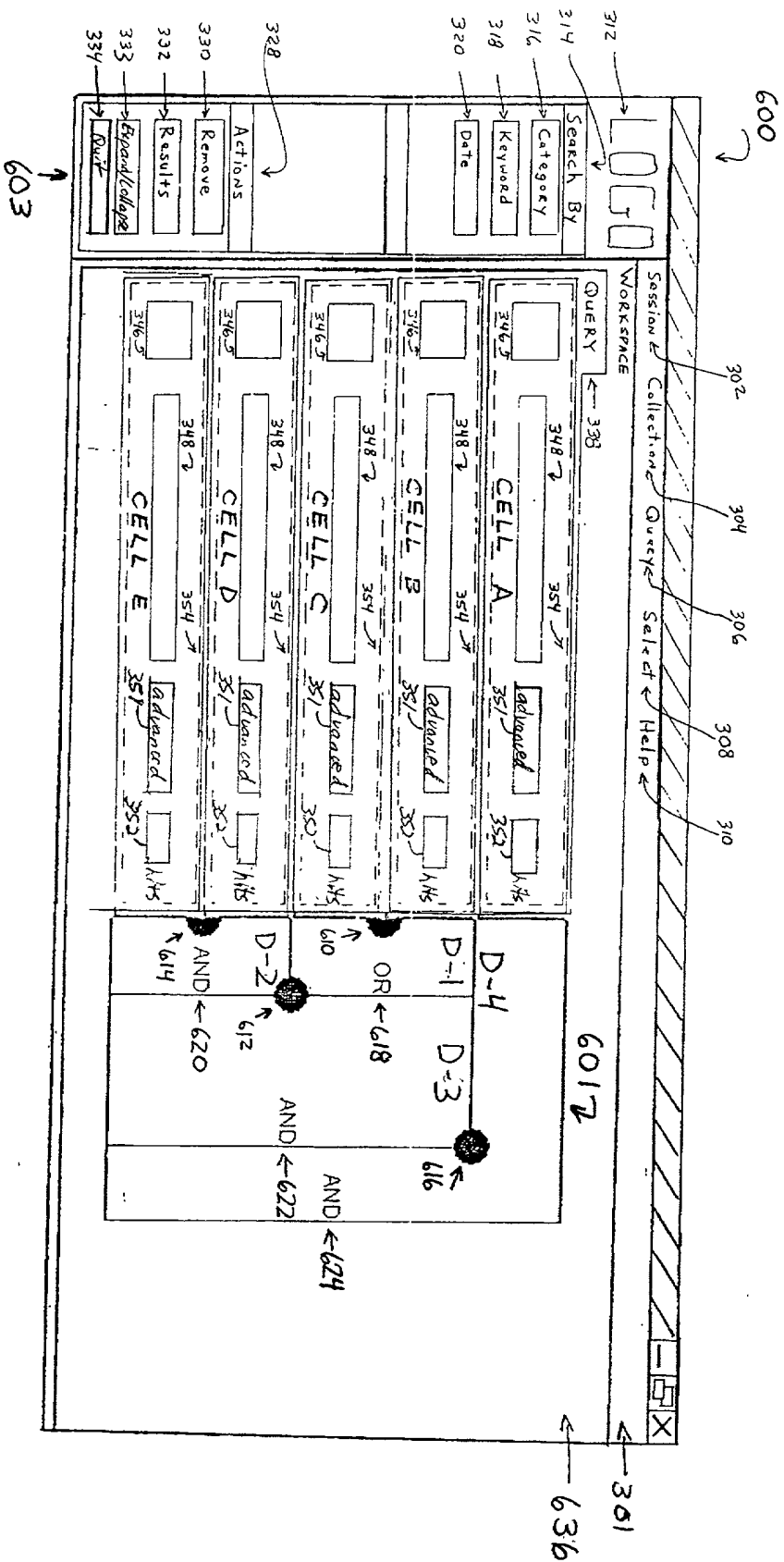


FIG. 6



626 → A[(B+C)(DE)]

FIG. 7

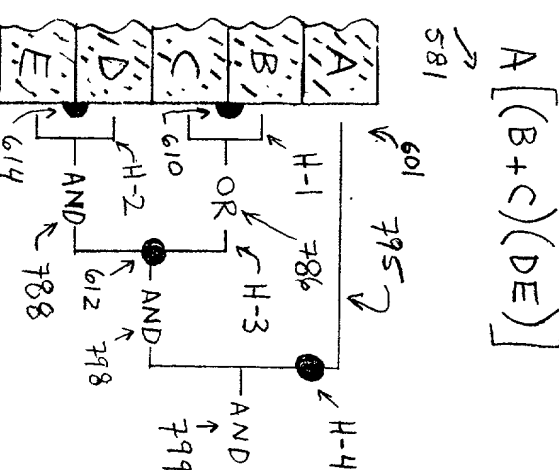
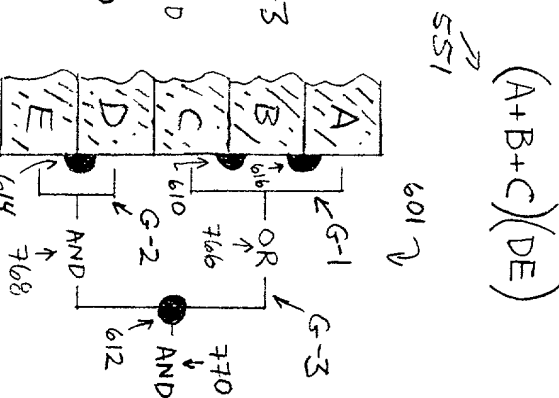
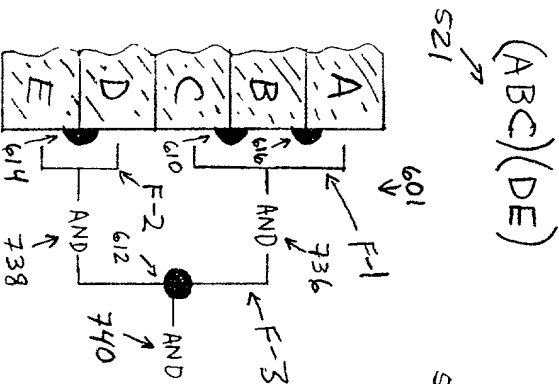
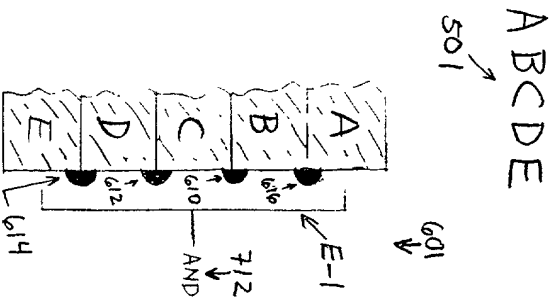
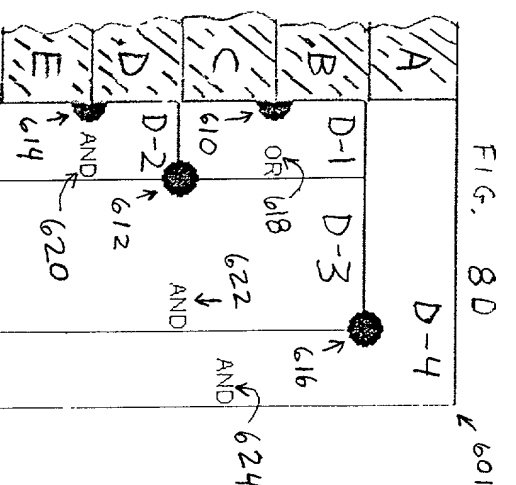
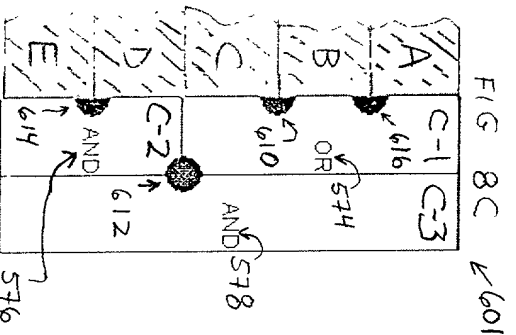
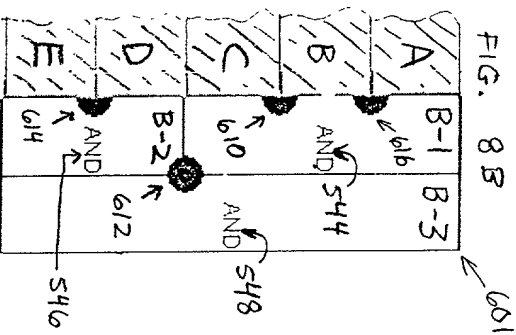
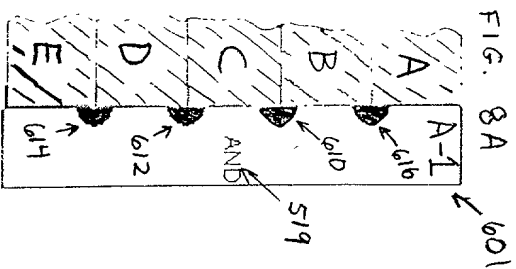


FIG. 8E

FIG. 8F

FIG. 8G

FIG. 8H

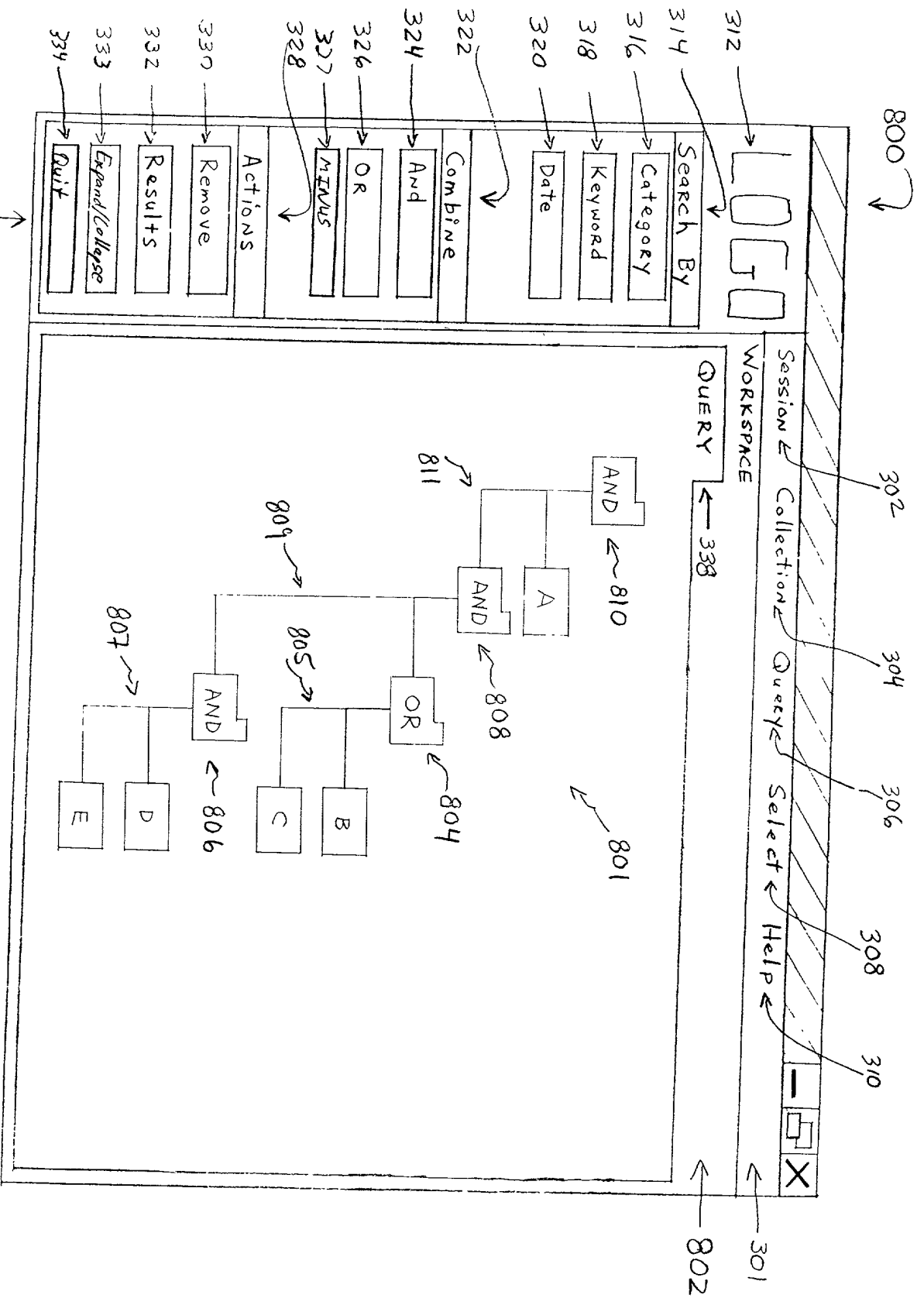
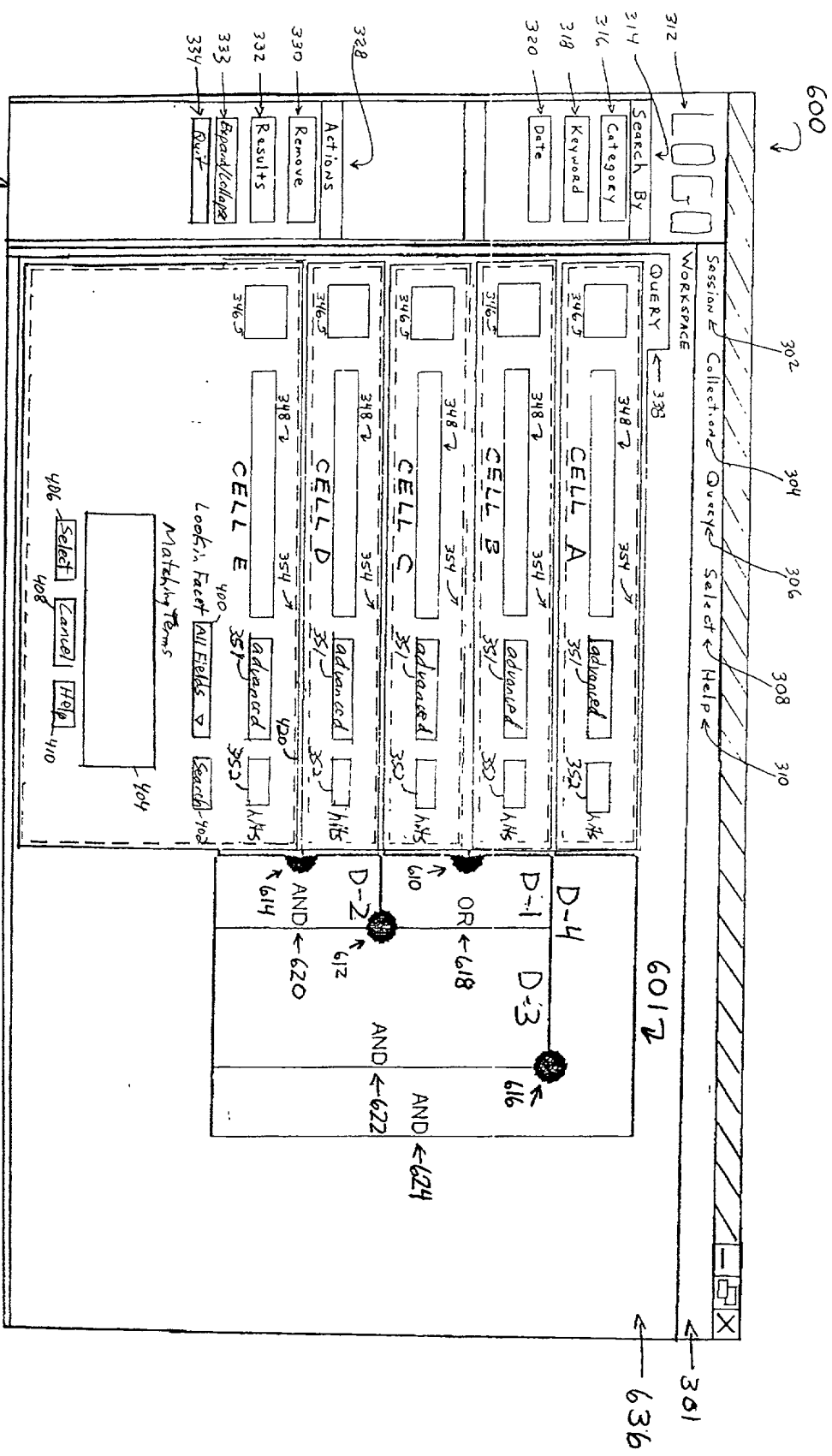


FIG. 9



603

626 -> A[(B+C)(DE)]

FIG. 10

FIG. 10 is a block diagram of a computer system 1000. The system 1000 includes a processor 1010, a memory 1020, a storage device 1030, and a network interface 1040. The processor 1010 is connected to the memory 1020, the storage device 1030, and the network interface 1040. The memory 1020 is connected to the storage device 1030 and the network interface 1040. The storage device 1030 is connected to the network interface 1040. The network interface 1040 is connected to a network 1050. The system 1000 is configured to execute a program 1060. The program 1060 is stored in the memory 1020. The program 1060 is executed by the processor 1010. The program 1060 causes the system 1000 to perform a set of operations 1070. The operations 1070 are performed by the processor 1010. The operations 1070 include receiving data 1080 from the network 1050, processing the data 1080, and outputting the results 1090 to the network 1050. The results 1090 are outputted to the network 1050 via the network interface 1040. The results 1090 are received by a client 1100. The client 1100 is connected to the network 1050. The client 1100 is configured to receive the results 1090 from the network 1050. The client 1100 is configured to display the results 1090 to a user 1110. The user 1110 is a person who interacts with the client 1100. The user 1110 is configured to provide input 1120 to the client 1100. The input 1120 is provided to the client 1100 via the network 1050. The input 1120 is received by the client 1100 via the network interface 1040. The input 1120 is processed by the processor 1010. The processor 1010 causes the system 1000 to perform the operations 1070 in response to the input 1120. The operations 1070 are performed by the processor 1010. The operations 1070 include receiving data 1080 from the network 1050, processing the data 1080, and outputting the results 1090 to the network 1050. The results 1090 are outputted to the network 1050 via the network interface 1040. The results 1090 are received by the client 1100. The client 1100 is connected to the network 1050. The client 1100 is configured to receive the results 1090 from the network 1050. The client 1100 is configured to display the results 1090 to a user 1110. The user 1110 is a person who interacts with the client 1100. The user 1110 is configured to provide input 1120 to the client 1100. The input 1120 is provided to the client 1100 via the network 1050. The input 1120 is received by the client 1100 via the network interface 1040. The input 1120 is processed by the processor 1010. The processor 1010 causes the system 1000 to perform the operations 1070 in response to the input 1120.

COMBINED DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY

As a below-named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated next to my name. I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: **GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS**. The specification of which is attached hereto. I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, Section 1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, § 119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)	Priority Claim	
	Yes	No
	<input type="checkbox"/>	<input type="checkbox"/>
Number	Country	Day/Month/Year Filed

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, § 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

Application Serial No.	Filing Date	Status: Patented, Pending, Abandoned
------------------------	-------------	--------------------------------------

I hereby appoint the following as my attorneys with full power of substitution to prosecute this application and transact all business in the Patent and Trademark Office in connection therewith:


Thomas B. Haverstock	32,571	Jonathan O. Owens	37,902
----------------------	--------	-------------------	--------

Please direct all correspondence regarding this application to the following:

Jonathan O. Owens
HAVERSTOCK & OWENS LLP
260 Sheridan Avenue, Suite 420
Palo Alto, California 94306

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of First Joint Inventor: Richard S. Neale

Inventor's Signature: 

Date: 6/12/00

Residence: 1260 Albion Lane, Sunnyvale, California 94087

Citizenship: United States

Post Office Address: 1260 Albion Lane, Sunnyvale, California 94087

Full Name of Second Joint Inventor: Loren L. Hart

Inventor's Signature: Loren L Hart ^{LH} 2006-05-02
Date

Residence: 4254 Indio Drive, San Jose, California 95136

Citizenship: United States

Post Office Address: 4254 Indio Drive, San Jose, California 95136

Full Name of Third Joint Inventor: Charlton D. Rose

Inventor's Signature: _____
Date

Residence: 13263 Mount Logan Street, Reno, Nevada 89506

Citizenship: United States

Post Office Address: 13263 Mount Logan Street, Reno, Nevada 89506

COMBINED DECLARATION FOR PATENT APPLICATION AND POWER OF ATTORNEY

As a below-named inventor, I hereby declare that

My residence, post office address and citizenship are as stated next to my name. I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled. **GRAPHICAL USER INTERFACE FOR BUILDING BOOLEAN QUERIES AND VIEWING SEARCH RESULTS**. The specification of which is attached hereto. I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, Section 1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, § 119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)

Priority Claim

Yes No

--	--

Number

Country

Day/Month/Year Filed

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, § 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application.

Application Serial No.

Filing Date

Status: Patented, Pending, Abandoned

I hereby appoint the following as my attorneys with full power of substitution to prosecute this application and transact all business in the Patent and Trademark Office in connection therewith:

Thomas B. Haverstock

32,571

Jonathan O. Owens

37,902

Please direct all correspondence regarding this application to the following:

Jonathan O. Owens
HAVERSTOCK & OWENS LLP
260 Sheridan Avenue, Suite 420
Palo Alto, California 94306

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full-Name of First Joint Inventor: Richard S. Neale

Inventor's Signature: _____

Date

Residence: 1260 Albion Lane, Sunnyvale, California 94087

Citizenship: United States

Post Office Address: 1260 Albion Lane, Sunnyvale, California 94087

Full Name of Second Joint Inventor Loren L Hart

Inventor's Signature _____ Date _____

Residence 4254 Indio Drive, San Jose, California 95136

Citizenship United States

Post Office Address 4254 Indio Drive, San Jose, California 95136

Full Name of Third Joint Inventor Charlton D Rose

Inventor's Signature Charlton D. Rose 6/29/00
Date

Residence 13263 Mount Logan Street, Reno, Nevada 89506

Citizenship United States

Post Office Address 13263 Mount Logan Street, Reno, Nevada 89506